CRANFIELD UNIVERSITY

ODYSSEAS KECHAGIAS STAMATIS

3D AUTOMATIC TARGET RECOGNITION FOR MISSILE PLATFORMS

DEFENCE ACADEMY

PhD
Academic Year: 2013 – 2016

Supervisor: Prof. N. Aouf
May 2017

CRANFIELD UNIVERSITY

DEFENCE ACADEMY

PhD

Academic Year 2013 - 2016

ODYSSEAS KECHAGIAS STAMATIS

3D AUTOMATIC TARGET RECOGNITION FOR MISSILE PLATFORMS

Supervisor: Prof. N. Aouf
May 2017

This thesis is submitted in partial fulfilment of the requirements for the degree of PhD

ABSTRACT

The quest for military Automatic Target Recognition (ATR) procedures arises from the demand to reduce collateral damage and fratricide. Although missiles with two-dimensional ATR capabilities do exist, the potential of future Light Detection and Ranging (LIDAR) missiles with three-dimensional (3D) ATR abilities shall significantly improve the missile's effectiveness in complex battlefields. This is because 3D ATR can encode the target's underlying structure and thus reinforce target recognition. However, the current military grade 3D ATR or military applied computer vision algorithms used for object recognition do not pose optimum solutions in the context of an ATR capable LIDAR based missile, primarily due to the computational and memory (in terms of storage) constraints that missiles impose.

Therefore, this research initially introduces a 3D descriptor taxonomy for the Local and the Global descriptor domain, capable of realising the processing cost of each potential option. Through these taxonomies, the optimum missile oriented descriptor per domain is identified that will further pinpoint the research route for this thesis.

In terms of 3D descriptors that are suitable for missiles, the contribution of this thesis is a 3D Global based descriptor and four 3D Local based descriptors namely the SURF Projection recognition (SPR), the Histogram of Distances (HoD), the processing efficient variant (HoD-S) and the binary variant B-HoD. These are challenged against current state-of-the-art 3D descriptors on standard commercial datasets, as well as on highly credible simulated air-to-ground missile engagement scenarios that consider various platform parameters and nuisances including simulated scale change and atmospheric disturbances.

The results obtained over the different datasets showed an outstanding computational improvement, on average x19 times faster than state-of-the-art techniques in the literature, while maintaining or even improving on some occasions the detection rate to a minimum of 90% and over of correct classified targets.

Keywords:

Light Detection and Ranging, Lightweight Processing, Missile Engagement Scenarios, Missile Seeker Architecture, Modeling, Simulation

ACKNOWLEDGEMENTS

I would like to offer my sincere gratitude to my supervisor and mentor Prof. Nabil Aouf whose restless supervision, motivating discussions and depth of knowledge accompanied me throughout this research quest. Thank you also for the support that you always cordially offered whenever needed. You made the process even more enjoyable.

In addition, I would like to thank MBDA UK for funding this research and especially Andy Sherriff and Gary Matson. I thank each of you for all your help and insightful comments.

I'd also like to thank Lounis Chermak, David Nam and Hamid Isakhani for their valuable time spent to proofread this thesis.

Finally, I wish to extend my special thanks to my wife and children, who have always given me their endless love and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF PSEUDO CODES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

2D	Two Dimensional
3D	Three Dimensional
3DSC	3D Shape Context
ASM	Anti-Ship Missile
ATR	Automatic Target Recognition
AUC	Area Under Curve
B-HoD	Binary Histogram of Distances
BRAND	Binary Robust Appearance and Normals Descriptor
CAD	Computer Aided Design
CFAR	Constant False Alert Rate
CPU	Central Processing Unit
CVFH	Clustered Viewpoint Feature Histograms
EM	Electromagnetic spectrum
ESF	Ensemble of Shape Functions
FLANN	Fast Library for Approximate Nearest Neighbours
FN	False Negative match
FoV	Field of View
FP	False Positive match
FPFH	Fast Point Feature Histogram
FPGA	Field Programmable Gate Arrays
GAH	Geometric Attribute Histogram
GER	Germany
GOOD	Global Orthographic Object Descriptor
GPU	Graphics Processing Unit

GRF	Global Reference Frame
GSS	Geometric Scale Space
HoD	Histogram of Distances
HoD-S	Histogram of Distances - Short
HONV	Histogram of Oriented Normal Vectors
HPR	Hidden Point Removal
HV	Hypothesis Verification
ICP	Iterated Closest Point
IR	Infrared
ISAR	Inverse Synthetic Aperture Radar
ISS	Intrinsic Shape Signatures
KL	Kullback - Leibler distance metric
k-NN	k-Nearest Neighbours
KPQ	KeyPoint Quality
LADAR	Laser Detection and Ranging
LIDAR	Light Detection and Ranging
Local D1	Local D1 Shape Distribution
LRA	Local Reference Axis
LRF	Local Reference Frame
LSP	Local Surface Patches
MATLAB	Matrix Laboratory software
MBT	Main Battle Tank
mmW	millimetric Wave
mr	mesh resolution
NARF	Normal Aligned Radial Features
NNDR	Nearest Neighbour Distance Ratio

OGH	Oriented Gradient Histogram
OUR-CVFH	Oriented, Unique and Repeatable Clustered Viewpoint Feature Histograms
PCL	Point Cloud Library
PDE	Projection Density Energy
PFH	Point Feature Histograms
RANSAC	Random Sample Consensus
RoPS	Rotational Projection Statistics
RUS	Russia
SAR	Synthetic Aperture Radar
SDH	Spatial Distribution Histogram
SE	Shannon Entropy
SHOT	Signature of Histograms of Orientations
SI	Shape Index
SIFT	Scale Invariant Feature Transform
SPR	SURF Projection Recognition
SURF	Speeded Up Robust Features
TN	True Negative match
ToF	Time of Flight
TP	True Positive match
USA	United States of America
USC	Unique Shape Context
VFH	Viewpoint Feature Histograms
WWII	World War II

1 Introduction

SINCE the introduction of missiles in WW II, they have seen a rapid evolution from the likes of V1 through the contemporary eclectic range of missiles manufactured by the defence industry. Regardless of the missile type, such as Anti-ship, Anti-tank or Air-to-Air, all missiles have a common goal: to engage with the intended target and achieve a target kill. In the case of an un-occluded, un-cluttered and constant-pose target, the missile can easily engage with the desired target at a high probability rate. Nevertheless, in real world scenarios this is not trivial, as clutter and non-target objects will be within the missile's Field of View (FoV) obscuring the target. The target will perform evasive manoeuvres trying to escape and thus it will constantly change its pose to the missile seeker with or without using countermeasures. In addition to the afore mentioned challenging task, the missile itself is flying at an extremely high speed further convoluting the process of correct-target acquisition and rapid data processing.

This thesis investigates the potentials of exploiting computer vision concepts for the future missile platforms implementing 3D Automatic Target recognition (ATR). The missile's sensor that acquires the target data i.e. missile seeker, is considered to be a Light Detection and Ranging (LIDAR) device that provides the 3D ATR algorithm with raw geometric coordinates *(x,y,z)* of all objects within the sensor's FoV.

Broadly, military ATR can be correlated with the *object recognition* process of commercial applications. Even though the computer vision community has already addressed 3D object recognition, current state-of-the-art 3D approaches are suboptimal in the military context. This is because the adverse, dynamically

changing and unstructured battlefield environment that a missile must operate is heavily distorted (noisy), cluttered and occluded. Such constraining operation-environment characteristics along with the limited processing time and hardware technology further impedes the implementation of a simple computer vision algorithm onboard a missile system. Additionally, military applications might involve loss of human life or even fratricide and therefore high target recognition performance is mandatory, necessitating the research community to focus further on the contemporary ATR problem.

Although this research aims at developing lightweight 3D descriptors for future intelligent missile systems, the concepts presented here are also applicable in a variety of non-military time-critical 3D object recognition applications. Indicatively, the proposed 3D descriptors and ATR architectures are appropriate for a great range of time-critical complex systems for space, air, and ground environments, generally, the law-enforcement and research establishments. Test trials revealed the performance of the developed techniques for various scenarios in the order of increasing difficulty. The suggested solutions assume no prior knowledge of the scenario.

1.1 Background

WW II was the entry point to a new era of warfare. For the first time in history a new weapon was used, named the missile, that offered several significant advantages compared to the ammunitions during the time. Since then, missiles have continuously evolved, and today they are the synonym of modern warfare due to their firing range and explosive payload. One of the aspiring desires since WW II has been the capability of missiles being able to autonomously recognise the target in order to increase the missile's effectiveness against camouflage, concealment and deception techniques applied by the enemy. Furthermore, target selection can provide impact point accuracy, reducing collateral damage and fratricide.

1.2 Problem statement

Military ATR research involves operating in different spatial and data modalities such as 2D IR [1]–[3], mmW radar [4], [5], 2D Synthetic Aperture Radar (SAR) [6], [7] and Inverse SAR (ISAR) [8]. Latest trends include 3D laser based solutions [9]–[13] exploiting an active LIDAR sensor. Even though military oriented ATR can be quite widespread among several modalities such as visual band and IR, current [14] and upcoming missile seeker ATR algorithms [1], [2] operate in the 2D IR.

The standard, yet extremely time-consuming policy in 2D pattern recognition problems, is matching a database that has a collection of templates representing possible viewings of each potential target against the scene target. These viewings are encoded based on a description technique. The number of templates per target is inversely proportional to the invariance of the description technique used such as to bridge the gap between the template poses.

Current and upcoming 2D IR ATR approaches, although offering appealing features such as manageable computational complexity and storage memory requirements, suffer from:

a. Limited robustness in out-of-plane rotations. This drawback is compensated with a very large number of templates aiming at bridging the descriptor's invariance limitations and ultimately achieving robustness to 3D target rotation. In fact, Gray *et al.* [1], [15] in their successful infrared domain ATR, propose a database consisting of 12 azimuthal viewings of each of the four naval targets to be recognised. In total, they use a database of 48 viewings on which localised target description based on the SIFT [16] technique is applied. Extending this strategy to achieve full 3D rotation invariance, would demand 123 viewings per target (12 viewings per pitch, roll and yaw rotation) leading to 6912 different poses for the same number of targets. The SIFT description technique encodes the surrounding area of a set of keypoints i.e. points of interest that have distinct characteristics and are automatically selected by SIFT, that in the case of a low-resolution image would be at least 20 per target pose. Thus

the entire template database of SIFT type descriptions shall contain a list of 138,240 entries per target that must be matched with the ones detected in the scene. Even in the case of using highly efficient matching strategies the excessive size of the database prohibits real-time performance.

b. Existing missiles [18] and theoretical IR ATR solutions for missiles [1], [2], [15] are evaluated in open sea environments where the target can accurately be segmented from the background.

c. The history of the target [17] affects its thermal signature. This is linked to whether the target is still hot or has cooled down. Therefore, the local 2D features of the target, which are based on a temperature related texture, can have a great variation. This imposes an excessive number of templates to cover possible heat variations. Adding this requirement to the already large number of templates needed to compensate a 3D rotation invariance, the database size becomes massive.

d. The target's thermal signature is affected by the time of the day [18]. This refers to the target's heat difference when compared to its surrounding environment.

e. Current camouflage [19] and countermeasure techniques affect the recognition performance [1].

1.3 Aims and constraints

Based upon the current 2D ATR deficiencies, 3D ATR based missiles can have an improved weapon effectiveness against camouflage, concealment and deception techniques because the laser beam which will be the mean to acquire the 3D data enables penetration of sparse structures. In addition, the short wavelength in which lasers operate provides high-resolution data and the capability to acquire details of the target reinforcing recognition applications.

This research aims at developing 3D descriptors suitable for future missiles with ATR capabilities that accommodate a LIDAR sensor. Thus, these descriptors must be compact enough to satisfy the computation and storage memory

requirements of the platform (missile). Given the software tools and the developing platform used in this research i.e. MATLAB on PC, the processing time threshold is set at 500ms. For more information on the reasoning for this threshold the reader is referred to Appendix A. Additionally, the proposed descriptors must achieve high recognition performance exceeding 90% under various perturbations and rigid transformations[1].

Another major constraint encountered is the classified nature of real military-application scenarios which cannot be disclosed to the public. Therefore, this thesis uses synthetic scenarios to challenge the proposed descriptors against current state-of-the-art 3D descriptors suggested in the available literature. Trials are also done on popular non-military databases from the computer vision domain aiming at a direct comparison of the proposed algorithms against current literature proposals on standard datasets.

1.4 Thesis Contribution

The contributions of this research are:

a. A 3D descriptor taxonomy for each of the main descriptor classes, namely one for the Local and one for the Global. The contribution to the former is amending the existing taxonomy with information about the data origin and the typically required pre-processing stages, and, with regard to the Global based descriptor class, current literature does not propose any taxonomy. Therefore, this work suggests a complete taxonomy following the architecture and rationale of the Local descriptor's one.

b. A Range Image based descriptor that introduces a 3D to a multi 2D ATR problem solving algorithm that exploits concepts from the mature 2D object recognition domain.

c. A 3D Global based descriptor that combines statistical analysis with RADAR theory and the 3D to multi-2D ATR problem solving concept.

[1] Although the higher recognition rate the better, for the current research purposes which are more a feasibility study rather than a complete 3D ATR solution, a recognition rate threshold of 90% is set based on common military industry practises.

d. Three 3D Local based descriptors that are processing efficient and have smaller storage requirements. Two of these are floating-point while the other is extended into the binary domain.

e. An extension of the standard computer vision 3D ATR architecture to facilitate the missile based 3D ATR requirements that relies on a single template scheme.

f. A missile oriented 3D ATR survey that evaluates current and suggested 3D descriptors on simulated but highly credible air-to-ground missile engagement scenarios with the missile being under various obliquities, distances to the target, and atmospheric perturbations.

These contributions produced the following publications:

1. **O. Kechagias-Stamatis**, N. Aouf, and M. A. Richardson, “3D automatic target recognition for future LIDAR missiles,” IEEE Transactions on Aerospace and Electronic Systems, vol. 52, no. 6, pp. 2662–2675, Dec. 2016.

2. **O. Kechagias-Stamatis** and N. Aouf, “Fast 3D object matching with Projection Density Energy,” in 2015 23rd Mediterranean Conference on Control and Automation (MED), 2015, pp. 752–758.

3. **O. Kechagias-Stamatis** and N. Aouf, “Histogram of distances for local surface description,” in 2016 IEEE International Conference on Robotics and Automation (ICRA), 2016, vol. 2016–June, pp. 2487–2493.

4. **O. Kechagias-Stamatis,** N. Aouf and L. Chermak, “B-HoD: A Lightweight and Fast Binary descriptor for 3D Object Recognition and Registration.”, 4th IEEE International Conference on Networking, Sensing and Control (ICNSC), 2017, (*in press*)

5. **O. Kechagias-Stamatis** and N. Aouf, “Evaluating 3D Local Descriptors for Future LIDAR Missiles with Automatic Target Recognition Capabilities,” *Imaging Science Journal*, 2017 (under review).

6. **O. Kechagias-Stamatis**, N. Aouf, and D. Nam, “3D Automatic Target

Recognition for UAV Platforms," in *Sensor Signal Processing for Defence (SSPD2017)*, 2017 (under review).

1.5 Thesis Structure

In addition to this introductory chapter, this thesis comprises of six more chapters. The following paragraphs shortly introduce each chapter and provide an insight of the content to follow. For better perspicuity and coherence, each chapter is independent and self-explanatory, presenting a relevant individual literature review at the beginning of each chapter.

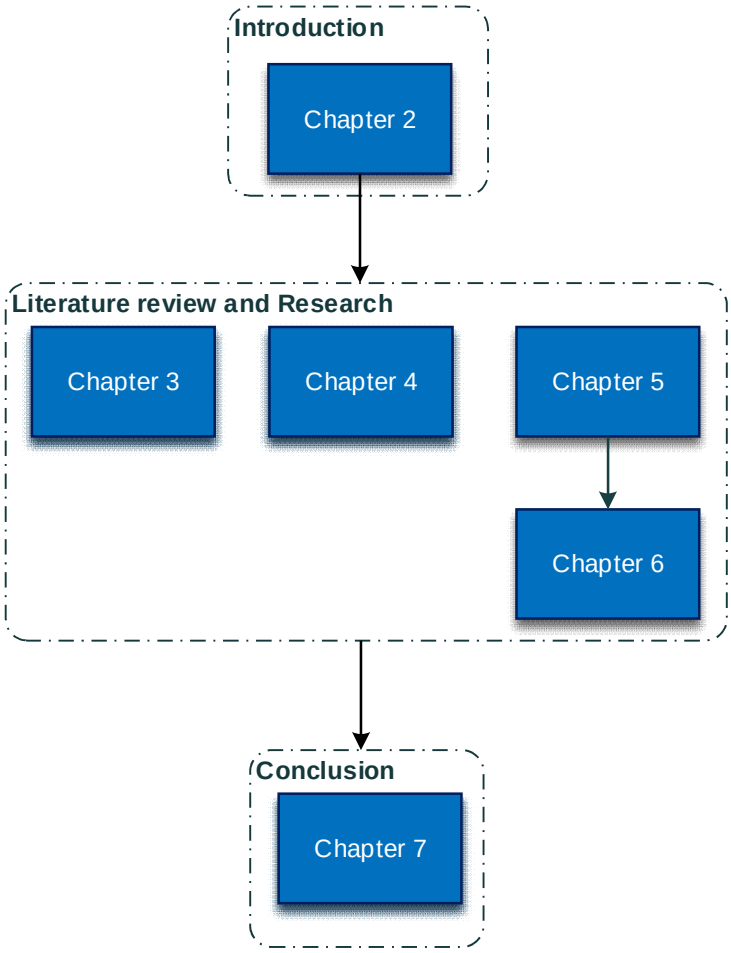


Figure 1- 1 Thesis layout

Chapter 2: 3D Automatic Target Recognition

This chapter sets the context of 3D ATR. Initially, it introduces the 3D data acquisition methods along with the advantages and limitations of 3D ATR. Then typical 3D ATR architectures are analysed i.e. keypoint detection, description, matching, hypothesis estimation and validation. The first contribution of this paper is the proposal of a local based 3D description taxonomy that complements the existing one and proposes an entirely new one for the Global based descriptors.

The same chapter presents a thorough analysis of current military oriented 3D ATR algorithms available in the open source literature, and, ultimately, a comprehensive list of computer vision based 3D descriptors is demonstrated.

Chapter 3: Range Image Based 3D ATR (paper 1)

This chapter introduces the contemporary range image based pattern recognition algorithms. Then the suggested technique is presented which extends the Speeded-Up Robust Features (SURF) method into the third dimension by solving multiple 2D ATR problems and performing template matching based on the extreme case of a single pose per target. Specifically, the proposed near-real time algorithm named SURF Projection Recognition (SPR) [20] transforms the 3D problem into multiple 2D projections of the 3D target on which 2D SURF is implemented. SURF matches per projection image are refined based on Hough pose clustering. SPR is robust against 3D rigid transformations combined with target subsampling and noise. Experiments on military targets from the Princeton shape benchmark and on a set of highly similar ground surface targets show that SPR provides high recognition rates in both cluttered and uncluttered scenarios.

Chapter 4: 3D Global based 3D ATR (paper 2)

This chapter analyses the Global 3D descriptors and highlights their deficiencies in the context of missile applications. Further, the proposed technique is thoroughly analysed which relies on the Projection Density Energy metric [13]

combined with a Constant False Alarm Rate adaptive threshold. Similar to the SPR algorithm proposed in Chapter 3, the 3D ATR problem is transformed into multiple 2Ds. The proposed PDE based approach is invariant to 3D rotations combined with scale change, Gaussian noise and target subsampling. Evaluation is performed on real targets from the UWA dataset and on military targets from the Princeton shape benchmark. Results indicate an appealing combination of high performance and a low processing time.

Chapter 5: 3D Local based 3D ATR (paper 3,4)

This chapter presents the current 3D local based descriptors along with their deficiencies with regard to missile applications. Current 3D local based descriptors, although accurate, their performance is restrained by the stability of their local reference frame or axis (LRF/A). Additionally, extra processing time is required to estimate the LRF/A of each local patch. In contrast to current trends, this chapter proposes a novel local based 3D descriptor entitled the Histogram of Distances (HoD) [21] and its computationally efficient variant HoD-S that override the necessity of a LRF/A and thus reducing drastically their processing time. The suggested descriptors encode the point-pair local distance distributions in multiple description and feature matching levels, providing fast to execute descriptors suitable for time-critical object recognition applications. Beyond computational efficiency, HoD and HoD-S are robust to severe noise levels and non-uniform subsampling. Evaluation on high, medium and low-quality popular point clouds suggests its promising performance compared to current state-of-the-art descriptors. A second contribution extends HoD into the Binary domain [22] aiming at reducing the storage memory requirements and matching time furthermore.

Chapter 6: Trials on Military Scenarios (paper 5,6)

In this chapter, a novel remote sensing targeting solution appropriate for future LIDAR active seeker missiles with 3D ATR capabilities is demonstrated [23], [24].

Specifically, it introduces an ATR pipeline that incorporates several pre and post-processing operations that extend the current computer vision architecture to facilitate the missile ATR requirements. Trials involve evaluation of HoD and HoD-S along with current state-of-the-art 3D local based descriptors, on several simulated but highly credible air-to-ground missile engagement scenarios. These military scenarios cover a variety of circumstances where the missile is under different obliquities, distances to the target and the scene is under various noise levels, subsampling levels, and atmospheric perturbations. Under these conditions, the recognition performance gained by the HoD and HoD-S descriptors are highly promising even in the extreme case of reducing the database entries to a single template per target.

Chapter 7: Conclusion

This chapter concludes this research by presenting a summary of the contributions amended with the proposed future work.

1.6 Software Tools

All algorithms and pipelines are developed in MATLAB software. This includes all feature descriptors and ATR architectures, the Hough pose clustering, the CFAR adaptive threshold, the Projection Density Energy estimation, the 3D to multiple 2D Projection transformation and the Correspondence-grouping algorithm.

C++ implementations of current 3D descriptors are obtained from the PCL library and are linked to the MATLAB pipeline via a MEX wrapper.

2 3D Automatic Target Recognition

THREE-dimensional target recognition shares many common features with 3D object/ pattern recognition. Their eminent difference is the operating environment, i.e. military vs. commercial and the platform constraints, such as the processing power and storage capacity.

Compared to classic 2D object recognition, 3D can afford superior recognition performance due to the unique features that a 3D representation has such as revealing the underlying structure of an object, illumination invariance, robustness to rotation, enhanced geometrical (depth) information and improved object pose estimation capability (an analysis is presented in Section 2.2). That advantageous recognition performance in combination with the low cost commercial type 3D data acquiring devices such as the *Microsoft Kinect*, the *Bumblebee XB3* and the *Asus Xtion Pro*, has increased the research interest in developing 3D pattern recognition algorithms. In fact, based on the available 3D object recognition literature, a cumulative research interest plot is created and shown in Figure 2-1 highlighting the constantly increasing number of 3D object descriptors. The graph presented in Figure 2-1 shows the sum of the available 3D descriptors from the first 3D descriptor in 1992 up to 2016. Additionally, the importance of 3D object recognition can be realised via the numerous applications in the fields of:

a. Robotics

   3D object detection and recognition [25], [26], 3D model classification [27]–[29], scene perception and reasoning [30]–[32].

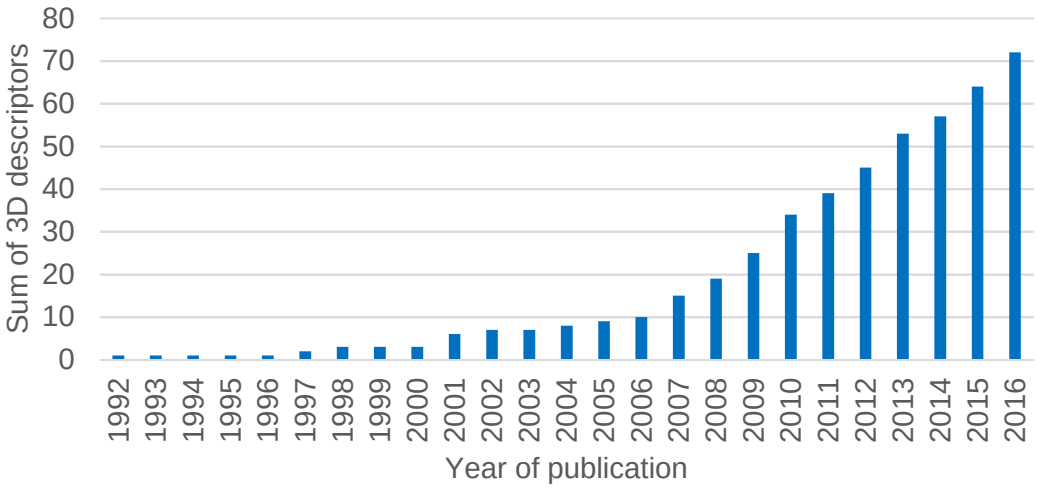


Figure 2- 1 Accumulative sum of 3D descriptors

b. Medical domain

   Multi-modality 3D medical image registration [33], computerised tomography X-rays analysis [34], ultrasound imaging [35], detection of brain morphological abnormalities [36] and 3D brain analysis [37], [38].

c. Industrial

   3D object detection in industrial environments [39].

d. Biometrics

   Face [40], [41], face expression [42] and ear [43] recognition.

e. Remote sensing

   Photogrammetry [44], ground scene registration [45], maritime vessel recognition [46]–[48].

f. Computer vision applications

   3D object registration [49]–[52], object detection, recognition and matching [53]–[64], 3D object classification [65] and 3D object retrieval [66].

g. Law enforcement / security

Airport baggage inspection [67]–[69].

h. Military

Target detection and recognition in ground [12], [70]–[75] and maritime environments [72], Human – machine interface via 3D Virtual reality [76].

2.1 3D data acquisition methods

Technical advancements allow acquiring 3D data by various means depending on the nature of the application. Most recent taxonomy of such techniques [77] amended with the latest technical achievements is presented in Figure 2- 2.

The basic categories are related to the sensor-target distance and can be distinguished into contact and contactless. The aim of this research is 3D ATR for missile applications and therefore contact-based equipment is not applicable. The contactless category is further subdivided into passive and active devices. The former is based on the stereoscopic effect by creating 3D data from stereo/ multiple 2D imagery. Although passive methods do not betray the position of the platform and therefore are highly desirable for military applications, a missile's diameter is not adequate to gain a valuable depth estimation for the 3D data. This happens due to the small baseline distance, which affords a depth estimation of less than 0.5 meter that is far less than the depth required. For further details please refer to Appendix B.

Active methods include three categories, namely the time-of-flight, the triangulation and the structured light based techniques. The latter two are not appealing options as they are mainly for short range controlled environments and therefore, the only viable option is the time-of-flight (ToF) acquisition method.

The ToF technique is conceptually equal to the RADAR principle but it exploits a different part of the EM spectrum. Specifically, a ToF based device transmits a laser or light pulse towards the target and measures with high precision the time it requires to return to its source i.e. the time-of-flight of the laser/ light pulse. Given the speed of light c, the roundtrip distance in meters between the laser source and the target is presented in Equation 2-1:

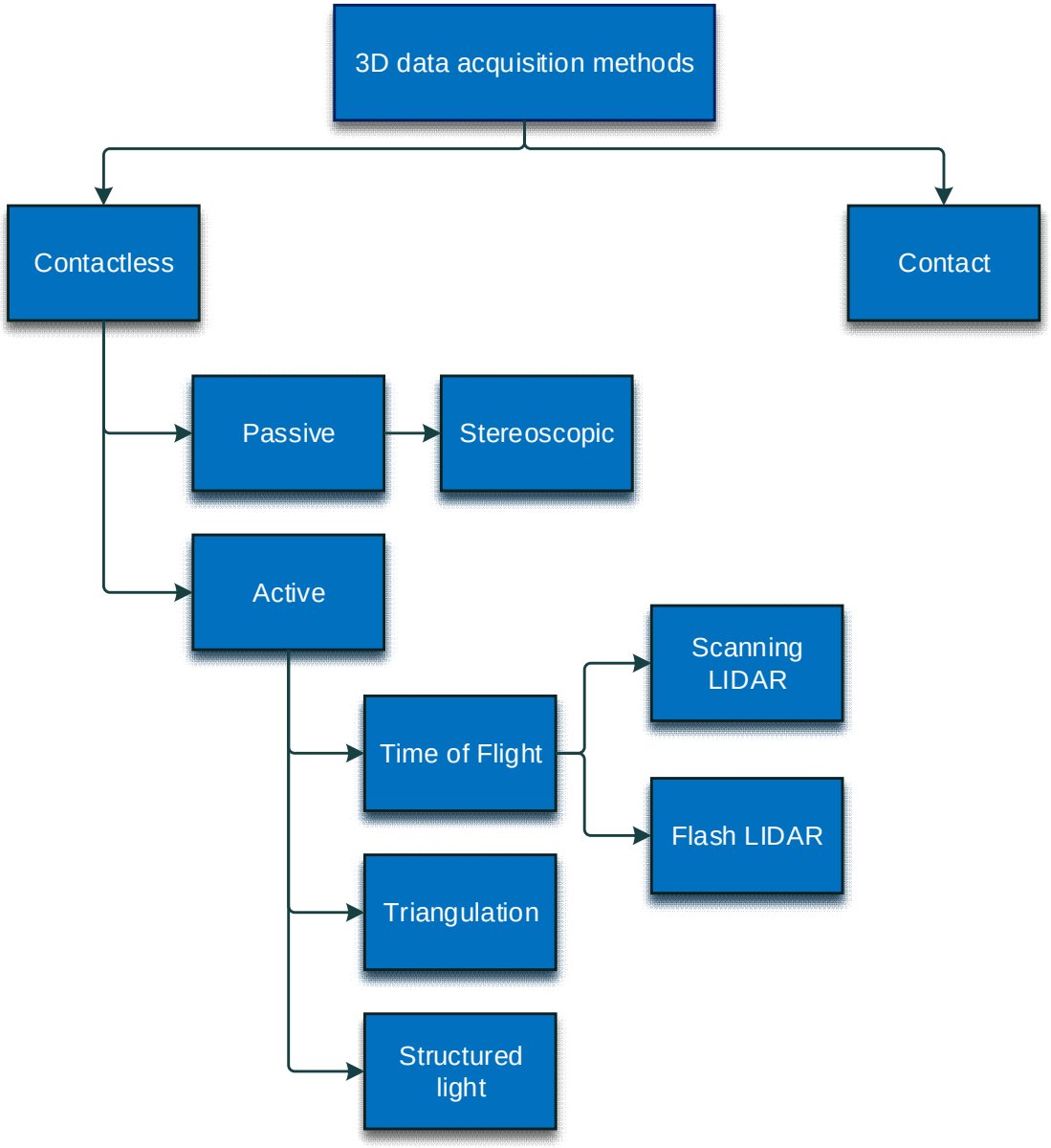


Figure 2- 2 Taxonomy of 3D data acquisition methods

$$d = \frac{c \cdot ToF}{2} \qquad \textbf{(2- 1)}$$

From these range measurements, a detailed 2D range/ depth estimation of the scene is created which can be converted into a 3D representation.

If the ToF device relies on laser pulses it uses the acronym LADAR (Laser Detection And Ranging), while if the device exploits any light source with the acronym LIDAR [78]. Since this research aims for a generic 3D ATR solution that relies on the depth information of the scene regardless of a laser or a light source, the term LIDAR will be used throughout this work.

Similar to the IR sensor distinction, LIDAR devices can either comprise of scanning or starring arrays. The former transmit bursts of laser/ light pulses that follow a predefined scanning pattern which is achieved by mechanically diverging these laser/ light pulses through a mirror.

Technical advancements have introduced a starring LIDAR array named 3D Flash LIDAR that relies on a single laser pulse covering the entire scene within the LIDAR's FoV. 3D Flash LIDARs can be parallelised to unconventional 2D digital cameras, as they have a 2D focal plane array with the difference being that this array can obtain the 3D ToF based distance and intensity information. For further information on the operating principles, the reader is referred to [79]. Figure 2- 3 presents the operating principles of both ToF type LIDAR devices.

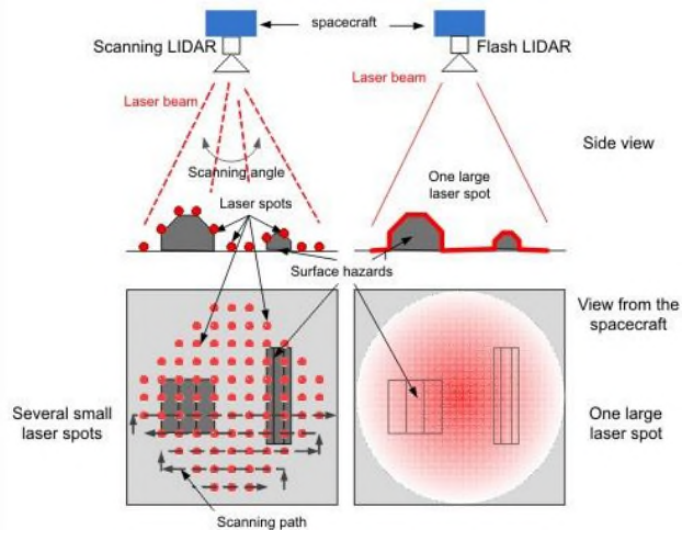


Figure 2- 3 Operating principle of Scanning and Flash LIDAR (image from [80]**)**

2.2 Advantages and limitations of 3D ATR

3D object recognition is an active research area as it offers numerous advantages over its 2D counterpart. Indicatively, 3D data exploit the geometric properties and the underlying structure of an object. These are more informative compared to 2D image information [57]. Also, features (data) extracted from the 3D domain are less affected by external illumination variation and target pose changes [41], [81]. Further appealing properties include robustness to rotation, enhanced geometrical (depth) information and improved object pose estimation capability [57].

With respect to future LIDAR based missiles, 3D ATR can afford all the advantages of 3D object recognition and therefore a LIDAR based missile shall have an improved effectiveness against camouflage, concealment and deception techniques. In addition, the short wavelength in which laser scanners operate provides high-resolution data and thus the capability to acquire details of the target reinforcing recognition applications. Finally, 3D ATR can offer accurate missile terminal guidance with aim point selection. These attractive features can enhance the ATR capability and reduce false alarms of future LIDAR missiles.

Although 3D LIDAR data has numerous advantages, it nevertheless has some disadvantages:

a. The number of photons (light energy) reflected depends on the spectral reflectance of the target for the corresponding laser/ light wavelength. Hence, a low reflective target or parts of the target might not reflect enough laser energy to trigger the LIDAR's receiver threshold and therefore these parts will be undetectable by the sensor. On the other hand, highly reflective targets (or parts of the target) at a close LIDAR sensor – target range might create dynamic pseudo artefacts i.e. lens flare.

b. 3D LIDARs are not as mature as other types of sensors e.g. visual based cameras and IR sensors. This mainly affects the operating range, data density, accuracy and data acquiring rate of such devices.

c. Currently, processing 3D data imposes a great computational burden compared to 2D data. Depending on the algorithm and the pre-processing required, 2D approaches can be at least one order of magnitude faster to execute.

d. The cost of such equipment is considerably higher compared to 2D visual or IR based sensors. Despite that, continuous technical development and market demand will reduce the production cost. This is already evident as low-cost LIDARs are already in the market [82].

e. LIDARs can be affected by sunlight. Despite that, some theoretical work has already been done to overcome that problem [83].

2.3 3D data representation

A LIDAR provides raw and unstructured data in a $\{(x,y,z)\,|\,x,y \in \mathbb{R}, z \in \mathbb{R}^{+}\}$ coordinate arrangement, as obtained from the scanning pattern. For completeness, *x, y* denotes the 2D coordinates relative to the LIDAR's boresight and is *z* the distance value between the LIDAR and the pointwise object in the scene that the laser/ light pulse has reflected on (Figure 2- 4).

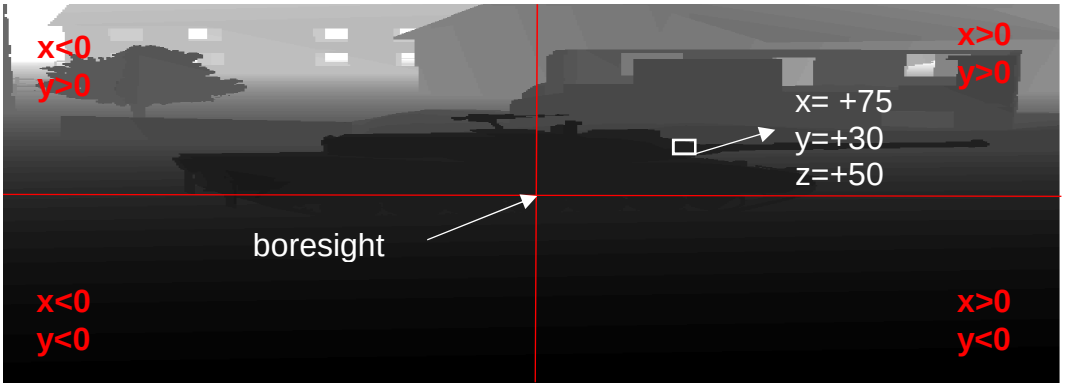


Figure 2- 4 Raw LIDAR data

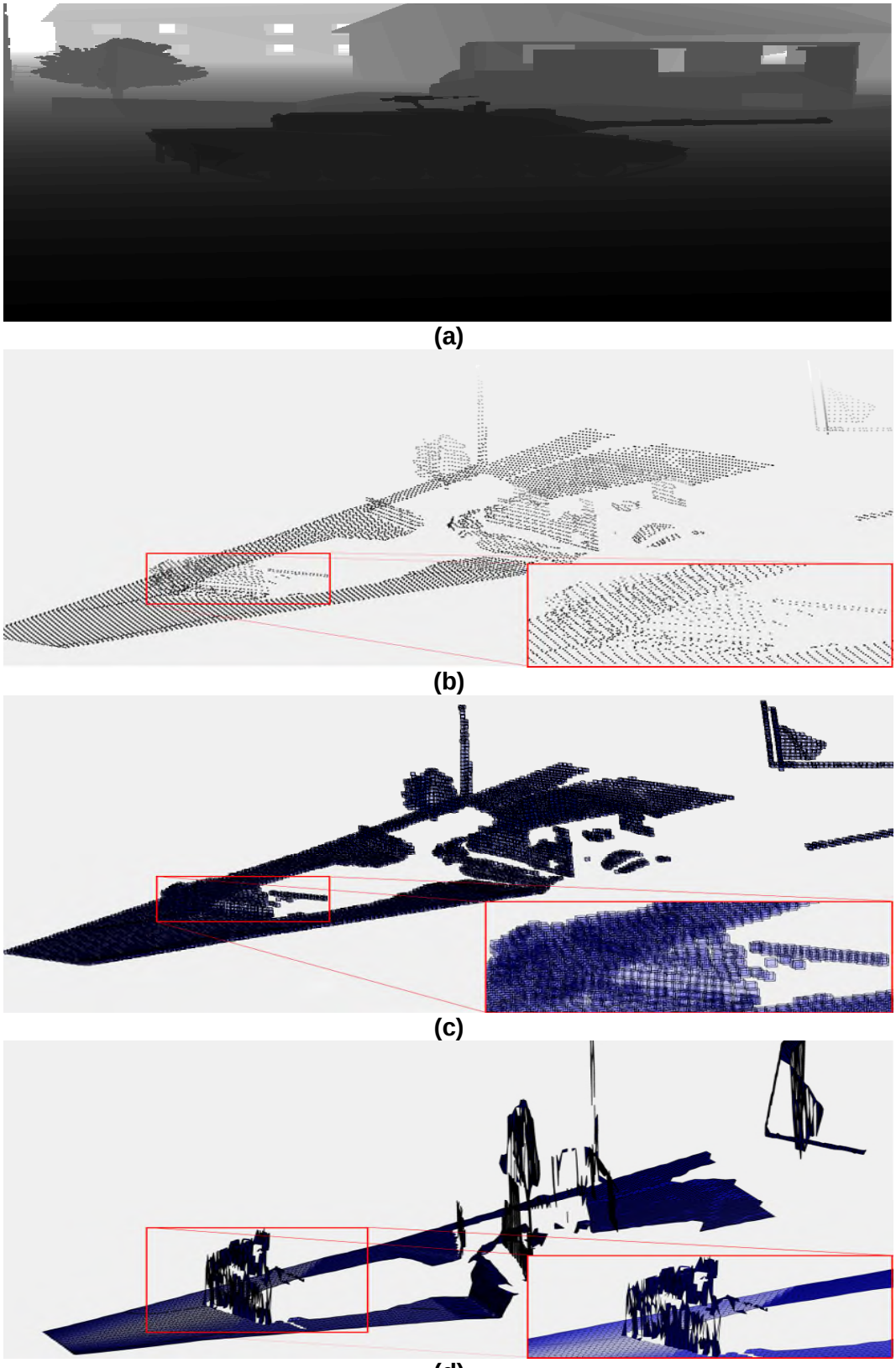
Representation of 3D data is either in a 2.5D or in a 3D form:

a. 2.5D images or range maps

   This is the simplest and most processing efficient way of representing 3D data as the *(x,y,z)* coordinates are simply converted into a 2D image form. An image is defined as a matrix where each cell, named pixel, stores a number representing the depth information z of the captured reflected laser/ light pulse. The latter can be visually presented as a colour variation within a colour band. In that way, for a grey colour band, 3D data would be shown as a 2D image where the grey intensity value relates to the depth/ range of pointwise object to the LIDAR device. Figure 2- 5 (a) shows a 2.5D image of a military scene, in a grey colour band where darker area is closer to the LIDAR.

b. Point clouds

   Another computationally efficient way of manipulating 3D data is by exploiting directly the *(x,y,z)* coordinates. In that case, 3D data are in the form of a cloud of vertices/ points at position *(x,y,z)* within an *XYZ* coordinate system centred at the LIDAR device. Figure 2- 5 (b) presents the point cloud equivalent version of the same target scene. More complicated ways of expressing 3D data include a voxel or a point cloud mesh. The former concerns transforming the raw *(x,y,z)* point cloud into a volumetric form by substituting the vertices of the raw point cloud with 3D voxels i.e. cubes. A voxel is the 3D volumetric equivalent to the 2D pixels and depending on its size, it may contain several vertices. Figure 2- 5 (c) presents a *voxelised* point cloud. On the other hand, for the meshes, *(x,y,z)* data are triangulated and the point cloud is converted into a set of connected triangular faces. An advantage of meshes is that they are more informative to raw point clouds as they include interconnections of the vertices (Figure 2- 5 (d)). Although compared to raw point clouds, voxels and meshes contain more geometric information, their creation requires extra processing time that affects the overall computational performance. An analysis is presented in section 2.4.2.4.



(a)

(b)

(c)

(d)

Figure 2- 5 3D data representation of a military scenario
(a) 2.5D depth map (b) 3D point cloud (c) 3D voxel (d) 3D mesh

2.4 Current 3D pattern recognition architectures

Regardless of the form that 3D data are in, i.e. 2.5D image or point cloud including its variants, a pattern recognition pipeline has the following core architecture. Initially, each template object is encoded based on a description technique. Then, the same description technique is applied to the scene object, and the scene descriptors are challenged against the template ones and possible matches i.e. correspondences are identified. For the 2.5D case, the template object that provides most matches gives its label to the target within the scene.

For the point cloud case, depending on the application and the matching accuracy required, several pre and post-processing applications such as template – target transformation hypothesis generation and verification might take place aiming at refining correspondences and reducing the number of mismatches. For completeness, point clouds can be described either globally (as one entity) or locally (in parts). Chapter 4 and 5 explicitly presents both these options. Figure 2- 6 shows a typical pattern recognition block diagram of the Local and Global architectures. An analysis of each block is presented in the following paragraphs.

Throughout this thesis singular entities such as point cloud vertices or keypoints will be presented with capital italics e.g. *P*, while clusters of these such as point clouds, with bold capital italics e.g. ***P***.

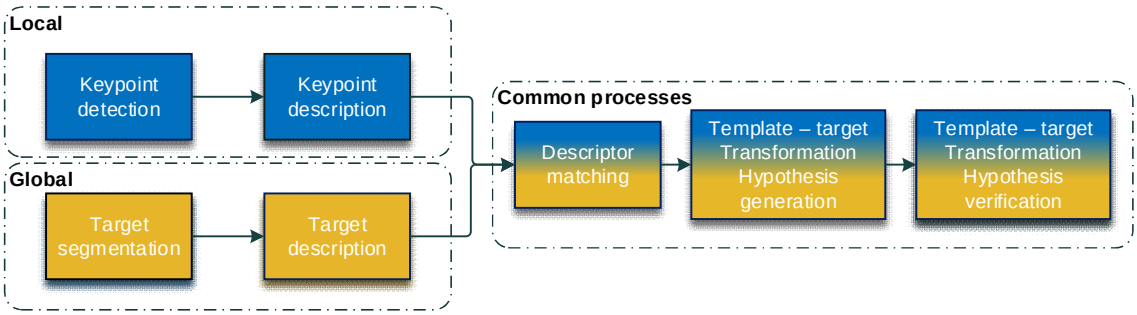


Figure 2- 6 Block diagrams of the Local and Global pattern recognition architectures

2.4.1 3D keypoint detectors

A keypoint detector is an algorithm that analyses the structure around a vertex and if that structure fulfils some specific criteria, it characterises that vertex as a point of interest i.e. keypoint. Ultimately, keypoint detectors aim at selecting vertices that are prominent among their surroundings, have unique features and can be redetected even if the object they belong to is distorted or corrupted.

Literature suggests a number of 3D keypoint detectors such as:

a. Shape Indexed based [40]

   Vertices centred at regions that have a Shape Index [84] satisfying some constraints considered as keypoints.

b. Intrinsic Shape Signatures (ISS) [85]

   Given a vertex, a scatter matrix of its surrounding area is established, which then undergoes an eigenvalue decomposition process. The eigenvalues are reordered in a decreasing row $\lambda_1, \lambda_2, \lambda_3$ and their pairwise ratios are considered $(\lambda_2/\lambda_1, \lambda_3/\lambda_2)$. If both these ratios fulfil some threshold criteria, then the vertex is considered as a keypoint.

c. KeyPoint Quality (KPQ) [86]

   This concept is similar to ISS but based only on the λ_2/λ_1 threshold. A pre-requisite for KPQ is that the surrounding area of the keypoint is aligned to its canonical pose based on the principal directions of its scatter matrix.

Figure 2- 7 depicts examples of the above keypoint detectors.

The main advantage of a keypoint detector is selecting a small fraction of the total vertices that are distinct and repeatable in pose changes and external nuisances. Although this can reduce the feature matching time, keypoint detectors impose an extra processing burden and can be prone to perturbations like subsampling and noise. For a thorough 3D keypoint detector evaluation the reader is referred to [87].

(a) (b) (c)

Figure 2- 7 3D keypoint detectors (a) Shape Index based (b) ISS (c) KPQ (images from [87]**)**

Considering the scope of this research, which is 3D ATR for missile platforms, processing efficiency and recognition performance are of equal importance. Therefore, in the proposed 3D local feature based solution (Chapter 5) instead of using a keypoint detector, keypoints are sparsely sampled throughout the scene. For the purposes of this research, this methodology has two main advantages. Firstly, it does not add extra time for keypoint detection. In addition, feature matching is based on a Fast Library for Approximate Nearest Neighbours (FLANN) [88] structure to compensate the great amount of features obtained from the scene and the template. Secondly, this strategy avoids the detector's performance to influence the overall ATR capabilities of the descriptor [89].

2.4.2 3D keypoint descriptors

A keypoint descriptor is an algorithm that analyses some features of the vertices belonging to the support region i.e. neighbourhood, of a keypoint. Depending on the descriptor, these features can be metrics such as distances, angular variations, coordinates. Ultimately, keypoint descriptors aim at describing the support region of a keypoint in a repeatable and unique manner that is robust to distortion or corruption. Additionally, a good descriptor must be robust to external perturbations, descriptive and compact [90], robust to rigid transformation and scale changes [91] and finally should be computationally efficient during its construction and matching.

3D descriptors can be distinguished about the 3D data domain they are applied on, as already presented in Section 2.3, i.e. on the ones that are designed for 2.5D data type or for 3D data. 3D object description techniques can broadly be divided into global and local feature based. The Global ones process and describe the object as one entity, while local feature based techniques describe only a small region around a keypoint.

In the following paragraphs a short introduction of each descriptor type is presented, while for better readability a thorough literature review analysis of a selection of each descriptor type is performed in the relevant chapter, that is 2.5D descriptors in Chapter 3, 3D Global based descriptors in Chapter 4 and 3D Local based descriptors in Chapter 5.

2.4.2.1 2.5D description techniques

2.5D images named also as range or depth images are essentially 2D representations of 3D data where the sensor – target distance is regarded as texture variation.

The literature suggests ATR on 2.5D imagery either by exploiting state-of-the-art 2D descriptors or by dedicated descriptors that are specially designed for 2.5D images. The former descriptors include SIFT [92] and SURF [93] or the binary BRIEF [94], ORB [95], BRISK [96] and FREAK [97]. Since the 2D algorithms are designed for colour RGB based images, it is the norm to apply a pre-processing step on the 2.5D images to bridge the colour – depth modality gap.

Approaches that are specifically designed to operate on 2.5D images are the Local Surface Patches (LSP) [40], the Normal Aligned Radial Features (NARF) [98], the Histogram of Oriented Normal Vectors (HONV) [99], the Binary Robust Appearance and Normals Descriptor (BRAND) [100], [101], IndSHOT [102], Pang's multi 2D projections [103] and the Geometric Scale Space (GSS) [49]. Figure 2- 8 depicts the existing 2.5D descriptors along with the contribution of this research in this category, the SURF Projection Recognition (SPR) [20]. For completeness, a selection of these techniques is analysed in Chapter 3. Detailed information per descriptor is presented in Table 2- 1 (index Nº 61 – 71) which is

allocated at the end of this chapter for better readability.

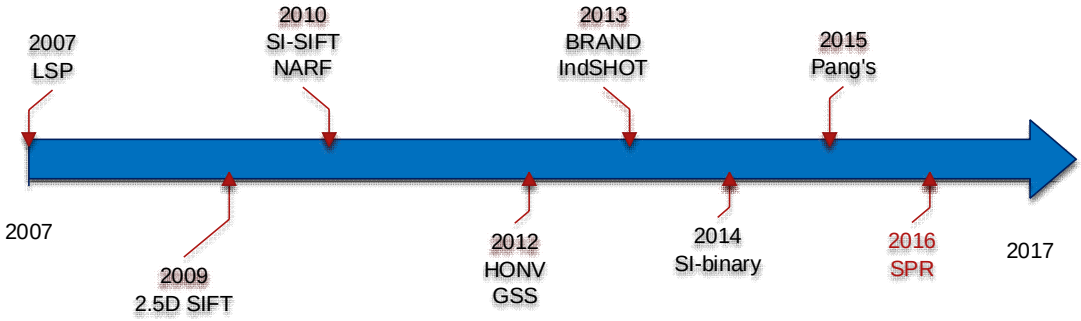


Figure 2- 8 Timeline representation of current Range Image based proposals

2.4.2.2 3D Global based description techniques

Techniques that belong to the Global description class process describe the object as one entity and have merely been used in 3D shape retrieval and classification [91]. Although their main advantage is computational efficiency [27], they demand *a priori* segmentation of the target from the scene [104] and are not robust against clutter and occlusion [53]. However, their processing efficiency is quite appealing for inter-class object recognition scenarios (target classification) and therefore they will be further analysed in this chapter under the scope of military ATR applications. Inter-class ATR refers to recognising objects belonging to different classes i.e. a fighter aircraft and a warship.

Examples of Global based techniques are the Shape Distributions [105], the Viewpoint Feature Histogram (VFH) [106], the Clustered VFH (CVFH) [28], the Oriented, Unique and Repeatable CVFH (OUR-CVFH) [32], the Ensemble of Shape Features (ESF) [27], the Compressed VFH [107], the 3D Feature Maps [108], the Geodesic Eccentricity method [109] and the Global Orthographic Object Descriptor (GOOD) [110]. The contribution of this chapter is the Projection Density Energy based (PDE) solution [13].

The existing Global 3D descriptors along with PDE are presented in Figure 2- 9. For completeness, a selection of these techniques is thoroughly analysed in

Chapter 4. Detailed information per descriptor is presented in Table 2- 1 (index Nº 52 – 60).

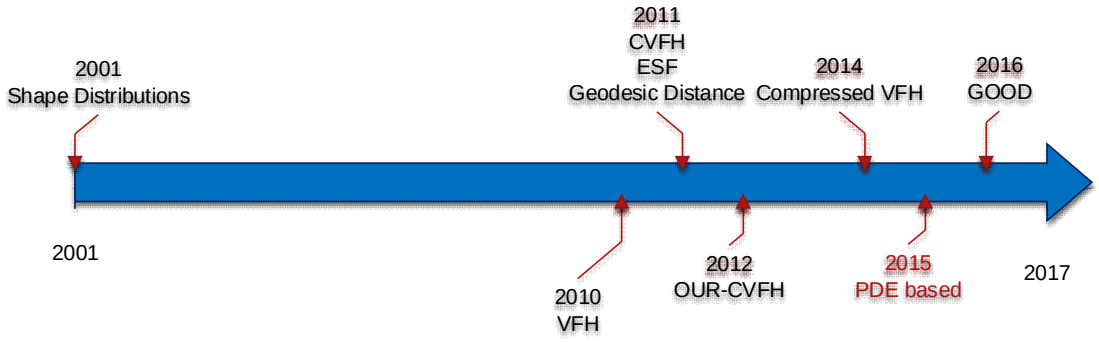


Figure 2- 9 Timeline representation of current Global 3D descriptors

2.4.2.3 3D Local based description techniques

Local feature based techniques describe local patches around a point of interest, i.e. support region, providing a valuable solution to partially visible objects in occluded scenes, in object registration, pose estimation and object recognition. Implementations are not restricted to computer vision or robotics applications but extend to navigation, remote sensing, industrial automation, biometrics, heath care, education, face recognition and military applications as presented in the introduction of Chapter 2. Due to the advantages of local 3D based descriptors, the literature suggests quite a few 3D local based descriptors of that type that are presented in Table 2- 1 (index No 1 – 51) that is located at the end of the current chapter (page 42). This table also includes the contributions of this research i.e. the Histogram of Distances (HoD), the HoD-Short (HoD-S), the Binary HoD (B-HoD) and the local D1 Shape Distribution (Local D1).

3D local based descriptors can be grouped based on whether their encoding process requires a Local Reference Frame (LRF), Local Reference Axis (LRA) or if they do not need a LRF/A. From Table 2- 1 it is obvious that the majority rely on a LRF, with the downside though the processing burden the LRF imposes (an analysis is presented in Section 5.1.7.1) and the fact that the robustness of these descriptors heavily relies on the repeatability and the accuracy of the LRF [111].

For completeness, a selection of these techniques based on their robustness and popularity is thoroughly analysed in Chapter 5.

2.4.2.4 3D descriptor taxonomy

Attempts creating a taxonomy of the local 3D descriptors are few, with the most recent being:

a. Tombari *et al.* [89], [112]

   The authors classify the 3D descriptors into three categories namely Signature, Histogram and Hybrid. The Signature class includes methods that describe the support region by measuring the geometric attributes such as normal and curvatures of a small surface patch in relation to a local coordinate basis. The latter basis is either a Local Reference Frame (LRF) or Axis (LRA). Although Signatures are very descriptive, minor noise and subsampling can highly affect the LRF/A estimation and thus the encoding of the support region itself. The Histograms class comprises of algorithms that describe the support region by clustering into histograms geometric or topological features such as mesh areas or number of vertices, based on a domain such as point coordinates, curvatures or normal angles. If the description domain is coordinate based, then methods belonging to this class are established on a LRF otherwise on a LRA. Although Histograms are less descriptive than Signatures, they are robust to noise and subsampling because small nuisances are compensated during the histogram type clustering. Finally, the Hybrid class combines attributes of both classes.

b. Guo *et al.* [57]

   Authors partially extend Tombari's taxonomy and propose a two-level classification. The first level includes the Histogram, Signature and Transform based methods with the Histogram and Signature being equal to Tombari's classification. For the Histogram class Guo *et al.* suggest a sub-classification layer that comprises of the Spatial Distribution Histogram (SDH), the Geometric Attribute Histogram (GAH) and the

Oriented Gradient Histogram (OGH). SDH describes the support region based on spatial distributed measurements e.g. number of vertices or mesh areas, which are then accumulated into a Histogram. GAH describes the support region based on its geometric attributes e.g. normal or curvature and OGH describes the support region based on oriented gradients of the support region. Finally, the Transform class includes methods that transform the 3D data from the spatial domain into another domain e.g. Voxel space, before initiating the description process.

A downside of both taxonomies is that they do not consider the originating data domain i.e. 2.5D image or point cloud and the typical pre-processing until the 3D descriptor is applied, but rather focus on the attributes of the descriptor. Driven by that, a complete 3D descriptor roadmap is suggested that includes three layers namely the data domain, pre-processing and final taxonomy. Figure 2- 10 depicts the suggested local 3D descriptor roadmap.

Selecting the most suitable 3D descriptor for a military application is not trivial, as it must balance performance and computational efficiency. Both these attributes are related to the data domain, pre-processing that might be required and the capabilities/ complexity of the descriptor itself. The advantage of the suggested taxonomy is that it can provide an insight of the attributes that a 3D descriptor has along with its rough computational requirements.

To support this, 100,000 unstructured *(x,y,z)* point coordinates are transformed in various forms i.e. raw 2.5D image, Shape Index 2.5D image, point cloud, 3D mesh and voxel of various leaf sizes i.e. number of points within each voxel. This trial aims at underpinning the processing burden of each data domain and processing required (Figure 2- 11).

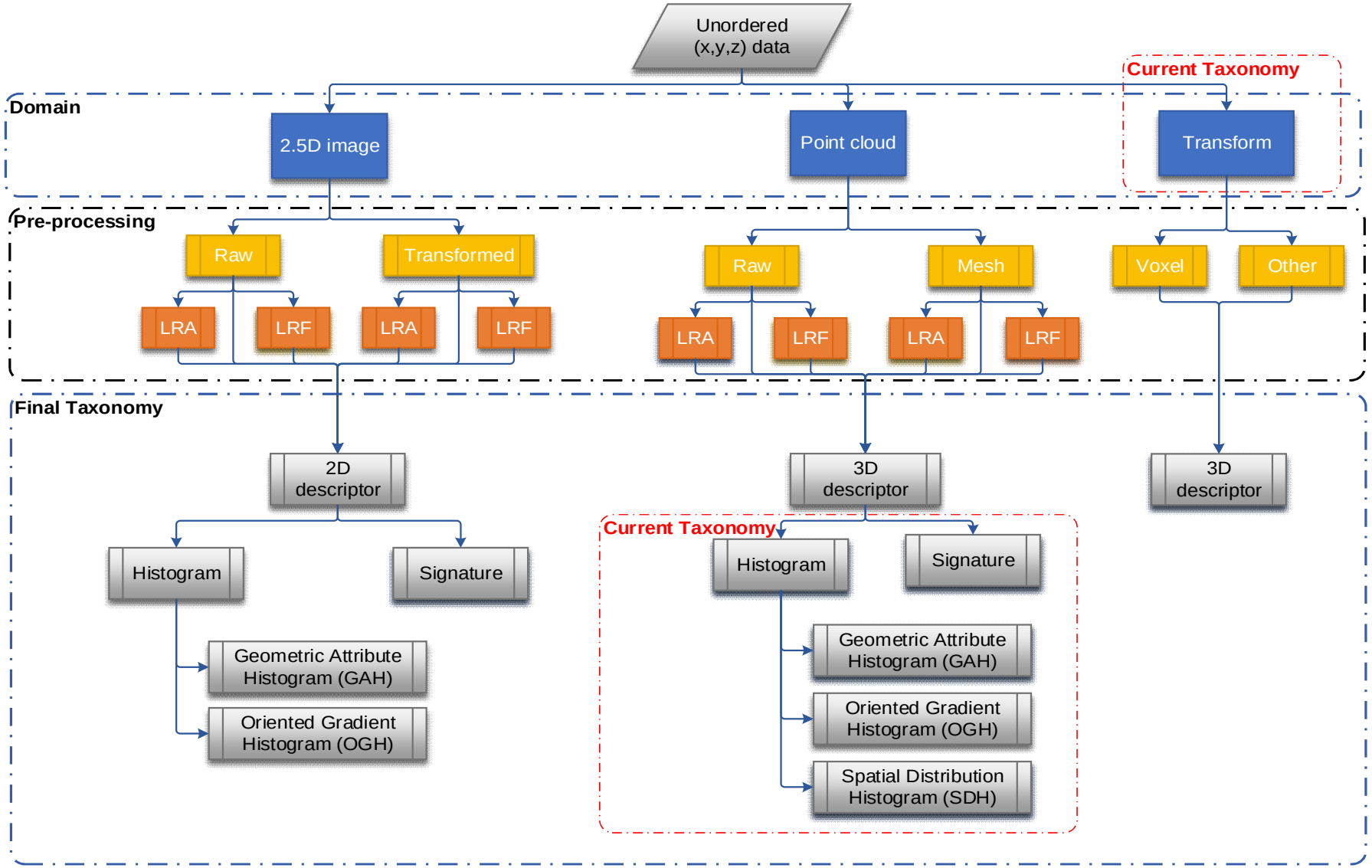


Figure 2- 10 Suggested local based 3D descriptor roadmap

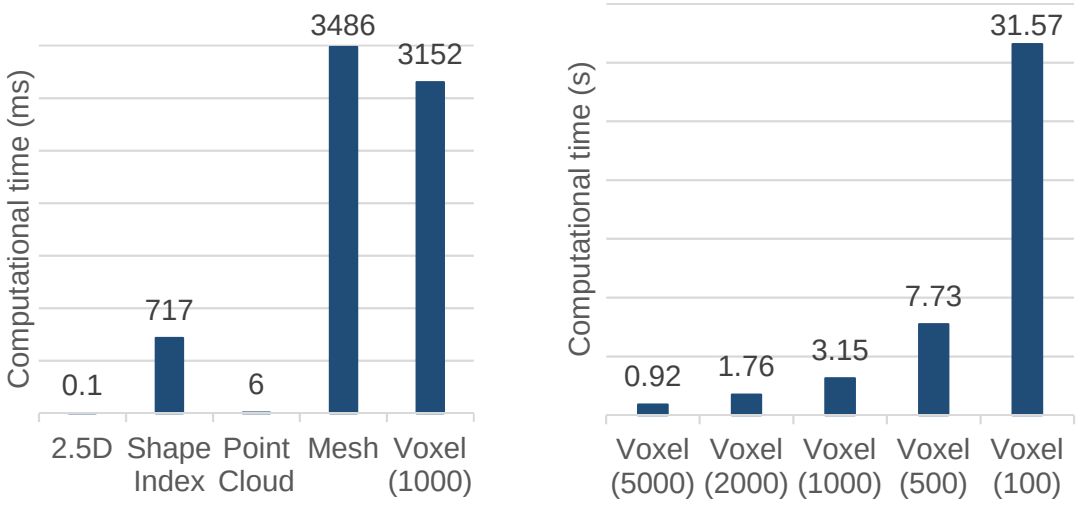


Figure 2- 11 (a) processing time per domain including data conversion (b) processing tie for various voxel sizes, value in brackets indicates the leaf size in points

As expected, by increasing the complexity, the computational burden increases. The importance of this experiment is identifying in specific the extra processing time required. Results obtained force this research to focus either on raw 2.5D images or raw point clouds. Indeed, converting raw data into a 2.5D image requires 0.1ms while into a point cloud structure 6ms. These two raw forms are several orders of magnitude faster compared to the rest of the competitor representations. An apparent downside is that both raw forms offer only spatial information rather than geometric or the inter-relationship of the vertices. This drawback must be compensated either by establishing a LRF/A or by designing a robust descriptor. Even though the former can boost a descriptor's performance [59], it increases the overall computational time and suffers from robustness to noise [113]. Hence, it is concluded that this research should focus on developing a local based descriptor in the:

a. 2.5D domain that uses raw data (Chapter 3)
b. Point cloud domain, based on raw data and without a LRF/A (Chapter 5)

Even though global based descriptors are inferior to the local ones for the reasons described in section 2.4.2.2, they still provide an appealing solution for object

classification and retrieval tasks. Current literature does not suggest any taxonomy for the global descriptors; therefore, Figure 2- 12 fills this gap by presenting a global 3D descriptor roadmap. Consistency among the global and the local roadmaps is preserved by sharing, where possible, the same structure. Considering the research rational of the 3D local feature description domain, the research about a global based 3D descriptor should be based on a raw 3D Point cloud neglecting a LRF (Chapter 4).

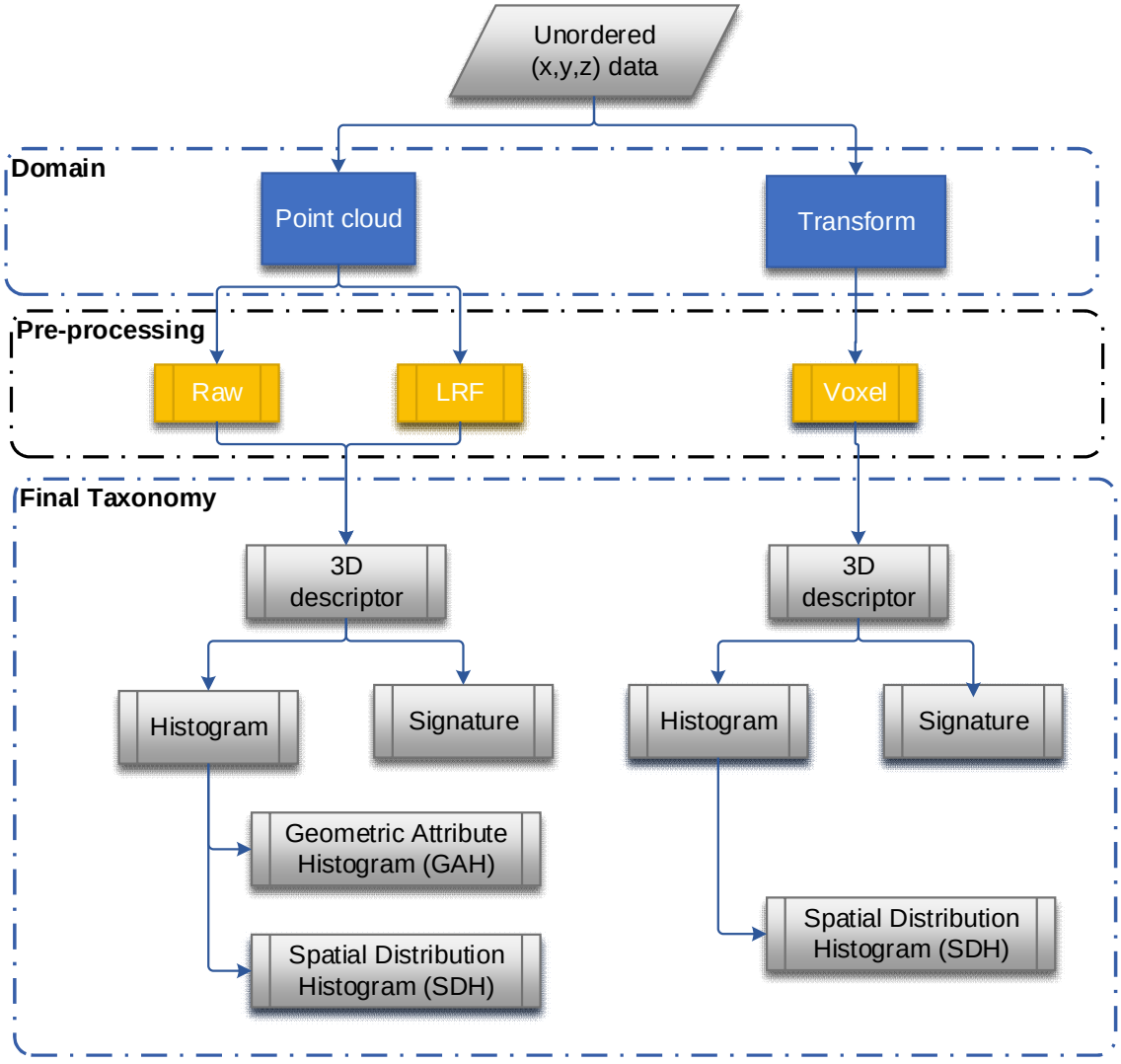


Figure 2- 12 Suggested global based 3D descriptor roadmap

2.4.3 Feature Matching, Hypothesis generation and verification

The target and the template descriptor are matched either by exploiting concepts from the 2D domain i.e. Nearest Neighbour Distance Ratio (NNDR) [114] or by relying on a cost function [104], [115]. Even if the descriptor is robust, the matching process might still produce some false correspondences. These can be reduced if the keypoints that the matched descriptor were extracted from are verified for their correctness.

Keypoint matching verification is done by initially estimating a transformation hypothesis between the scene and the target matched keypoints as obtained from the NNDR matching process. This transformation hypothesis is then applied on the template for a coarse scene – template alignment, followed by a fine alignment via the Iterative Closest Point (ICP) process. This procedure repeats for all qualified templates and the one that provides the smallest alignment error within specified limits is verified as being recognised within the scene template. Literature suggests several transformation Hypothesis generation and verification methods. For further details the reader is referred to [57].

2.4.4 Military oriented 3D Automatic Target Recognition

The battlefield is a noisy, highly cluttered and occluded, dynamically changing environment. These demanding features require the implementation of robust object recognition techniques capable to fulfil the needs of a missile platform with ATR capabilities. Based on open source military oriented ATR algorithms are based on Spin Images [10], geometric fitting [74], [75], the Baseline Processing Pipeline [9], and the parts-based articulated target recognition [116].

Although current military oriented 3D ATR proposals have interesting features, they have many drawbacks that prohibit implementing them on missile platforms. Their common feature is exploiting the standard pipeline of Figure 2- 6 at most up to the feature matching stage. Concepts and setbacks of open source military 3D ATR algorithms are presented in the following paragraphs.

2.4.4.1 Spin Images

One of the most cited local 3D descriptor is the Spin Image [64]. The raw point cloud $\boldsymbol{P} \in \mathbb{R}$ that consists of K vertices $P_a, \{a \mid a \in \mathbb{N}, a \leq K\}$ is transformed into a mesh. For each keypoint P_a acting as a centroid, a spherical volume $\boldsymbol{V}$ of radius r is extracted. For each $\boldsymbol{V}$ that contains the vertices $P_d, \{d \mid d \in \mathbb{N}, d \leq a\}$, the normal n is calculated which will define the z-axis of a LRA. Based on that LRA, P_d vertices are remapped from the Cartesian into a cylindrical coordinate system. Finally the Spin Image descriptor is based on accumulating the transformed P_d points enclosed within each bin of a rectangular grid that is rotated around the LRA axis. The grid and bin sizes determine the samplings of the local area.

Although Spin Images have been an appealing solution for quite a long time, they present several drawbacks:

a. They have low descriptiveness and are sensitive to mesh resolution changes [81], [85], [106]. The former disadvantage is due to information loss induced during the 3D to 2D coordinate remapping.

b. They have limited robustness to noise [117], occlusion and clutter [49], [55], [62].

c. Converting the point cloud into a mesh can be a time-consuming procedure, especially if the scene mesh cardinality is large.

d. Spin Images are not scale invariant [81] and are not robust to uniform [118] or non-uniform sampling [119].

e. Spin Images suffer from localisation errors of the keypoints [90]

f. Even though Spin Images have been used in target recognition [10], their performance has been investigated only in top-down viewing situations, where the target's features are more distinctive compared to the side view ones. In addition, this target pose is not always the case during a missile – target engagement scenario. An example is shown in Figure 2- 13.

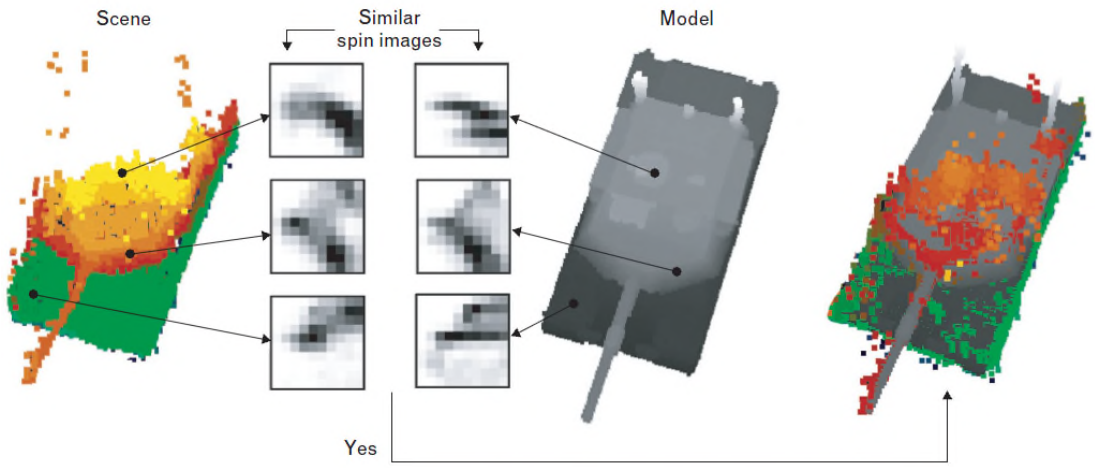


Figure 2- 13 Spin Images for (a) target (b) model (c) target and model alignment is based on the transformation hypothesis created from the matched template – target Spin Images (image from [10]**)**

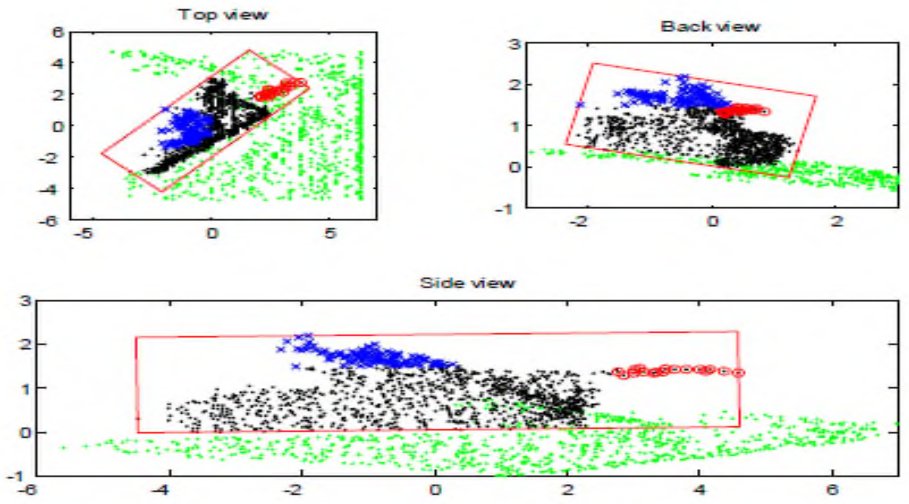
2.4.4.2 Geometric fitting

Geometric fitting [74], [75] decomposes the scene into a number of rectangle-based regions, based on the assumption that man-made objects are such. Geometric fitting is a two-staged algorithm:

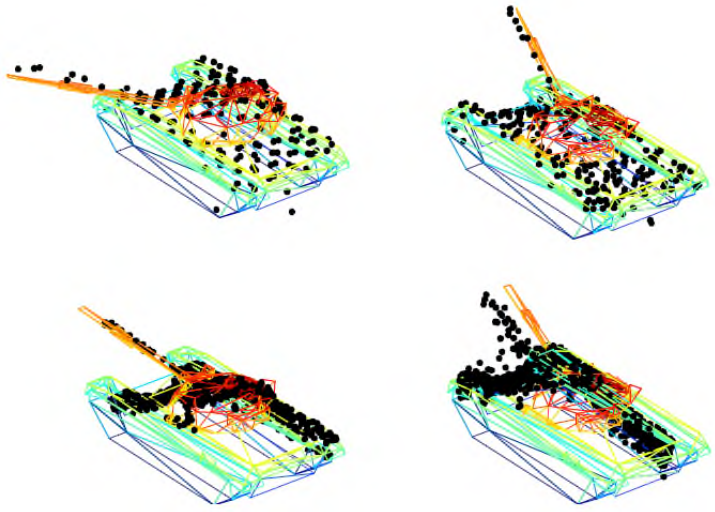
a. The first phase considers segmenting the target from the scene. Given a manually defined global reference frame (GRF), the target is projected on the three planes of the GRF and its size and orientation are estimated. The latter is done by solving a minimisation problem with the function being the area that encloses the detected keypoints. Then, the rectangle of each projection that encloses the target is divided into smaller non-overlapping rectangles based on their consistency. Finally, simple geometric comparisons are performed between the templates and the sub-rectangular parts to determine the ones that do not belong to any template. By rejecting a sub-rectangle, the scene vertices it contains are rejected as well.

The second phase considers target recognition. The scene vertices contained within the remaining rectangles are aligned with low resolution template CAD

models. The template keypoints that provide the smallest Euclidean distance based Mean Square Error with the scene points are considered as the template that is matched with the target within the scene. Figure 2- 14 presents a Geometric fitting based target recognition example.



(a)



(b)

Figure 2- 14 Geometric fitting based target recognition (a) target is enclosed within a red rectangle with the barrel samples and turret samples in red and blue respectively (b) point clouds of the target (in black) are aligned with the corresponding wire-frame low resolution CAD model. (images from [74], [75]**)**

Although this proposal has appealing features, it has the following drawbacks:

a. It assumes the target is already detected within the scene which is an already complex procedure that in real scenarios cannot be taken for granted.

b. It requires a GRF that has to be manually aligned to an almost flat ground surface. The first component is not applicable to autonomous platforms while the latter i.e. flat ground surface requirement, is not always the case in real-world scenarios.

c. Scenarios tested do not include clutter and partially occluded targets. In fact, it is expected that during the rectangle estimation phase, the clutter objects would interfere and so the entire algorithm would lack a good performance.

d. Minimisation problems require much time until they converge to a solution adding extra processing time to the entire process.

2.4.4.3 Parts-based articulated target recognition

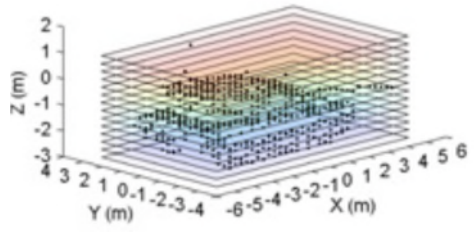
This solution is appropriate for MBT target recognition when its main components have an articulated rigid motion i.e. the turret and the hull are in a non-canonical position [116]. Initially the target's point cloud is projected onto the planes of a GRF coordinate system. For each plane, the normalised entropy is calculated, named as the Projection Density Energy (PDE). Considering that man-made objects are smooth and rectangular, the transition area between the hull and the turret is defined by a global PDE minimum. Based on the transition area, the target is decomposed into the turret and hull which are aligned in a canonical position and are then recomposed. Finally, the processed target is matched against the template by minimising the error obtained by the ICP algorithm.

Although this target recognition algorithm has the advantage of handling articulated targets, it has the following drawbacks:

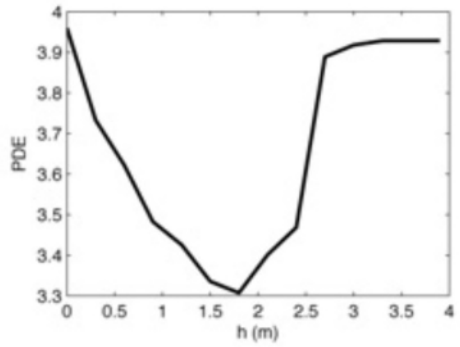
a. It cannot handle occlusion, noise and non-uniform subsampling as these will alter the PDE values and introduce pseudo transition areas.

b. The target must be segmented from the scene as clutter can interfere with the PDE estimation.

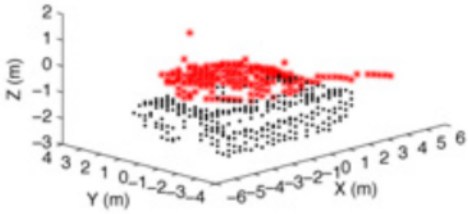
An example of this approach is presented in Figure 2- 15.



(a)



(b)



(c)

Figure 2- 15 Parts based articulated target recognition (a) Target point cloud is resampled (b) PDE with respect to height (c) target decomposition to hull and turret (images from [116]**)**

2.4.4.4 Baseline Processing Pipeline

The Baseline Processing Pipeline [116] estimates the ground level via the Random Sample Consensus (RANSAC) algorithm and then the vertices above the ground level are clustered into Volumes of Interest. These volumes are refined based on their physical dimensions i.e. width and length, and the remaining ones are qualified for the description stage named Target Geometry Mapping. The latter creates for each volume a 3D height map based on a user defined grid size. Descriptor matching is executed by calculating the euclidean

distance between the target and the template Target Geometry Matching features. An example of the Baseline Processing Pipeline approach is presented in Figure 2- 16.

This global based descriptor is effective only if all the following strict assumptions are fulfilled which is difficult to occur within a complex battlefield:

a. The scene has a planar ground.

b. Targets are un-occluded, without clutter and they are predominantly longer than they are wide.

c. Scenarios consider only a look-down case.

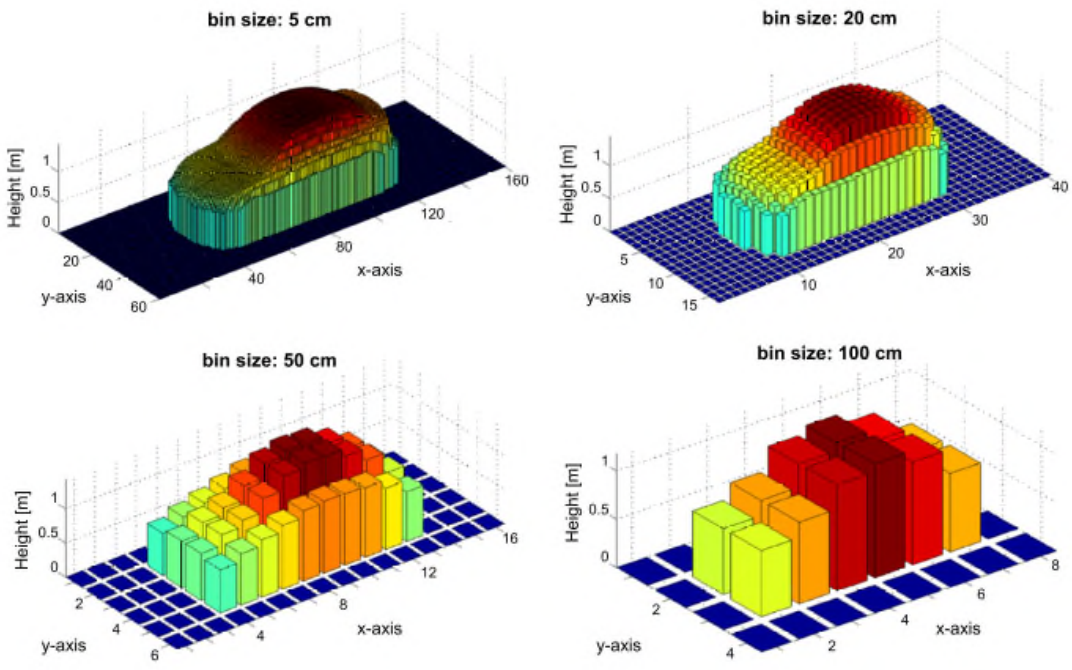


Figure 2- 16 Target Geometry Mapping with various grid sizes (images from [9]**)**

2.4.5 Computer vision based 3D Automatic Target Recognition descriptors

The computer vision community suggests a great number of 3D descriptors that encode an object/target in a global or a local manner exploiting the taxonomy

presented in Figure 2- 12 and Figure 2- 10 respectively. Current local descriptors achieve high quality recognition performance while the target is occluded, cluttered and under various perturbations such as noise and data subsampling. Although it would be interesting simply to transfer the available descriptors from the computer vision domain into a military context, this methodology is questionable mainly for computational inefficiency reasons and robustness to severe noise levels and non-uniform subsampling. An evaluation of current state-of-the-art 3D descriptors is presented in Chapters 3-6.

Table 2- 1 presents an extensive list of the available 3D descriptors, extended and amended [57] such as to facilitate the suggested taxonomies. 3D descriptors that are contributed by this research are highlighted in bold face. For better readability and coherence, the most appreciated/cited 3D descriptors along with the proposed descriptor per domain are analysed in the relevant chapter. That is, Chapter 3 for the 2.5D, Chapter 4 for the Global and Chapter 5 for the Local based.

2.5 Conclusion

In this chapter, the author introduced the basic concepts related to 3D Automatic Target Recognition ranging from 3D data acquisition methods to current computer vision 3D recognition architectures. Then, roadmaps for both the Local and the Global 3D descriptors were set, identifying not only the final taxonomy of each descriptor, but linking it to its originating domain and potential pre-processing operations.

Although the open literature offers a few military oriented 3D ATR algorithms, these are not appealing for missile platforms for the reasons presented in paragraph 2.4.4. On the other hand, the computer vision community offers a great range of 3D object recognition solutions (Table 2- 1) that will be evaluated in the following Chapters 3-6.

Table 2- 1 Computer vision based 3D descriptors

Nº	Name	Year	Category	Domain	Data Type	Reference Frame	Taxonomy
1	Splash [120]	1992	Local	3D	Mesh	LRA	Signature
2	Point Signasture [121]	1997	Local	3D	Mesh	LRF	Signature
3	Spin Image [64]	1998	Local	3D	Mesh	LRA	SDH
4	Point's Fingerprint [122]	2001	Local	3D	Mesh	LRF	Signature
5	Spherical Spin Images [123]	2001	Local	3D	Mesh	LRA	SDH
6	Surface Signature [124]	2002	Local	3D	Mesh	LRA	GAH
7	3DSC [55]	2004	Local	3D	Point Cloud	LRA	SDH
8	NBS [125]	2005	Local	3D	Mesh	LRA	Signature
9	3D Tensor [81], [126]	2006	Local	3D	Mesh	LRF	SDH
10	THRIFT [65]	2007	Local	3D	Point Cloud	NO	GAH
11	Snapshot [127]	2007	Local	3D	Mesh	LRF	Signature
12	VD-LSD [128]	2007	Local	3D	Point Cloud	NO	GAH
13	RIFT [119]	2007	Local	3D	Point Cloud	LRF	OGH
14	HMM [54]	2008	Local	3D	Mesh	NO	Signature
15	Hua's [37]	2008	Local	3D	Mesh	LRA	OGH
16	EM [129]	2008	Local	3D	Mesh	LRF	Signature
17	PFH [130]	2008	Local	3D	Point Cloud	LRF	GAH
18	Spectral Feature [131]	2009	Local	3D	Mesh	LRF	Transform
19	FPFH [117]	2009	Local	3D	Point Cloud	LRF	GAH
20	HKS [132]	2009	Local	3D	Mesh	NO	Transform
21	MeshHOG [133]	2009	Local	3D	Mesh	LRF	OGH
22	ISS [85]	2009	Local	3D	Point Cloud	LRF	SDH
23	Hou's [42]	2010	Local	3D	Mesh	LRF	OGH

24	3D SURF [134]	2010	Local	3D	Voxel	LRF	Transform
25	Depth Values [86]	2010	Local	3D	Point Cloud	LRF	Signature
26	SHOT [52], [112]	2010	Local	3D	Mesh	LRF	GAH
27	USC [89]	2010	Local	3D	Mesh	LRF	SDH
28	CORS [135]	2010	Local	3D	Point Cloud	LRA	Signature
29	CSHOT [136]	2011	Local	3D	Mesh	LRF	GAH
30	RSD [30], [137]	2011	Local	3D	Point Cloud	LRA	OGH
31	LD-SIFT [50]	2012	Local	3D	Mesh	LRA	OGH
32	ISC [138]	2012	Local	3D	Mesh	NO	SDH
33	SURE [139]	2012	Local	3D	Point Cloud	LRF	OGH
34	APSC [140]	2013	Local	3D	Mesh	LRF	SDH
35	TriSI [63]	2013	Local	3D	Mesh	LRF	SDH
36	RoPS [59], [61]	2013	Local	3D	Mesh	LRF	SDH
37	3D-Div [141]	2013	Local	3D	Point Cloud	LRF	OGH
38	3D Haar based [39]	2013	Local	3D	Voxel	LRA	Signature
39	C-RoPS [56]	2013	Local	3D	Mesh	LRF	SDH
40	PC-RoPS [45]	2014	Local	3D	Point Cloud	LRF	SDH
41	Multi scale RoPS [58]	2014	Local	3D	Mesh	LRF	SDH
42	IROPS [142]	2015	Local	3D	Mesh	LRF	SDH
43	B-SHOT [143]	2015	Local	3D	Point Cloud	LRF	GAH
44	CoSPAIR [25]	2015	Local	3D	Point Cloud	LRF	SDH
45	SUAH [144]	2015	Local	3D	Mesh	LRF	GAH
46	SIPF [145]	2015	Local	3D	Point Cloud	LRA	Signature
47	LFSH [113]	2016	Local	3D	Point Cloud	LRA	SDH
48	**HoD** [21]	2016	Local	3D	Point Cloud	NO	SDH
49	**HoD-S** [21]	2016	Local	3D	Point Cloud	NO	SDH

50	**B-HoD** [22]	2016	Local	3D	Point Cloud	NO	SDH
51	**Local-D1** [21]	2016	Local	3D	Point Cloud	NO	SDH
52	Shape Distributions [105]	2001	Global	3D	Point Cloud	NO	Signature
53	VFH [106]	2010	Global	3D	Point Cloud	LRF	GAH
54	CVFH [28]	2011	Global	3D	Point Cloud	LRF	GAH
55	ESF [27]	2011	Global	3D	Voxel	NO	SDH
56	Eccentricity based [109]	2011	Global	3D	Voxel	NO	Signature
57	OUR-CVFH [32]	2012	Global	3D	Point Cloud	LRF	GAH
58	Compressed VFH [107]	2014	Global	3D	Point Cloud	LRF	GAH
59	**Projection Density Energy** [13]	2015	Global	3D	Point Cloud	NO	SDH
60	GOOD [110]	2016	Global	3D	Point Cloud	LRF	GAH
61	LSP [40], [43]	2007	Local	2.5D	Transformed	NO	GAH
62	2.5D SIFT [146]	2009	Local	2.5D	Transformed	LRA	OGH
63	SI-SIFT [53]	2010	Local	2.5D	Transformed	LRA	OGH
64	NARF [31]	2010	Local	2.5D	raw	LRA	Signature
65	HONV [99]	2012	Local	2.5D	raw	LRF	GAH
66	GSS [49]	2012	Local	2.5D	raw	LRF	Signature
67	BRAND [100], [147]	2013	Local	2.5D	raw	LRA	Signature
68	IndSHOT [102]	2013	Local	2.5D	Transformed	LRF	GAH
69	SI-binary [148]	2014	Local	2.5D	Transformed	LRA	OGH
70	Pang's [103]	2015	Local	2.5D	raw	NO	OGH
71	**SPR** [20]	2016	Local	2.5D	raw	LRA	OGH

3 Range Image Based 3D ATR

RANGE images are 2D representations of 3D data where the sensor – target distance is regarded as texture variation. Since 3D data are in a 2D form while preserving 3D information, literature classifies range images as 2.5D images. As already presented in Section 2.4.2.1, literature suggests ATR on 2.5D imagery either by exploiting state-of-the-art 2D descriptors based on SIFT [92], SURF [93], BRIEF [94], ORB [95], BRISK [96] and FREAK [97] or by introducing descriptors designed for 2.5D images like LSP [40], NARF [98], HONV [99], BRAND [100], [101], IndSHOT [102], Pang's multi 2D projections [103] and GSS [49]. As a reminder, Figure 3- 1 shows current 2.5D descriptors and the suggested SURF Projection Recognition (SPR) [20]. For completeness, a selection of these techniques is analysed in the following sections.

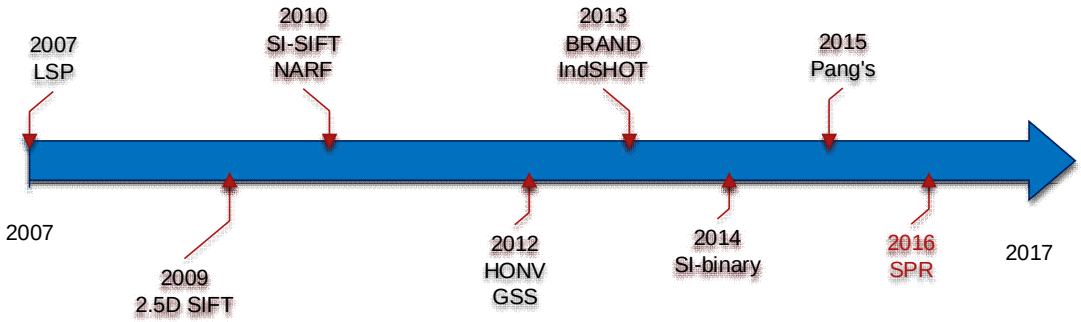


Figure 3- 1 Timeline representation of current 2.5D descriptors

3.1.1 2D based algorithms

This category applies conventional 2D description techniques onto the 2.5D imagery. These techniques aim at encoding the local oriented gradients on RGB imagery. Since 2.5D images have depth related texture that is usually quite smooth, it is the norm to add a pre-processing stage to enhance the range image information content. Such methods are extensively examined in [148] which suggests that transforming a depth image into its Shape Index (*SI*) form is the most effective option. This is because compared to the other enhancement methods examined in this paper, *SI* affords more keypoints to be detected by each keypoint detection method. This is appealing because face recognition, which is the scope of this paper, can gain higher recognition rates.

3.1.1.1 SIFT based proposals

Bayramoglu and Atalan [53] as well as Krizaj *et al.* [148] convert the raw range image into its *SI* form and then apply SIFT. The former authors name this technique SI-SIFT. *SI* [149] is based on the minimum and maximum normal curvatures of a local support region, namely the principal curvatures k_1 and k_2. Hence for a 2.5D image I, the $SI_{(i,j)}, \{i, j \mid i, j \in \mathbb{N}, i \leq w, j \leq h\}$ with w being the width and h the height of I is given by:

$$SI_{(i,j)} = \frac{1}{2} - \frac{1}{\pi}\arctan\left(\frac{k_{1_{(i,j)}} + k_{2_{(i,j)}}}{k_{1_{(i,j)}} - k_{2_{(i,j)}}}\right) \quad \textbf{(3- 1)}$$

where k_1 and k_2 are given by:

$$K_{1_{(i,j)}} = H_{(i,j)} + \sqrt{H_{(i,j)}^2 - K_{(i,j)}} \quad \textbf{(3- 2)}$$

$$K_{1_{(i,j)}} = H_{(i,j)} - \sqrt{H_{(i,j)}^2 - K_{(i,j)}} \quad \textbf{(3- 3)}$$

and the Gaussian and the mean curvature K and H respectively are:

$$H_{(i,j)} = \frac{\left(1+f_y^2\right)f_{xx} + \left(1+f_x^2\right)f_{yy} - 2f_x f_y f_{xy}}{2\left(\sqrt{1+f_x^2+f_y^2}\right)} \quad \textbf{(3- 4)}$$

$$K_{(i,j)} = \frac{f_{xx}f_{yy} - f_{xy}^2}{\left(1 + f_x^2 + f_y^2\right)^2} \qquad \textbf{(3- 5)}$$

where f_x, f_y, f_{xx}, f_{xy} denote the first and second order Gaussian derivatives at coordinates *(i,j)*. Typical *SI* representations are shown Figure 3- 2.

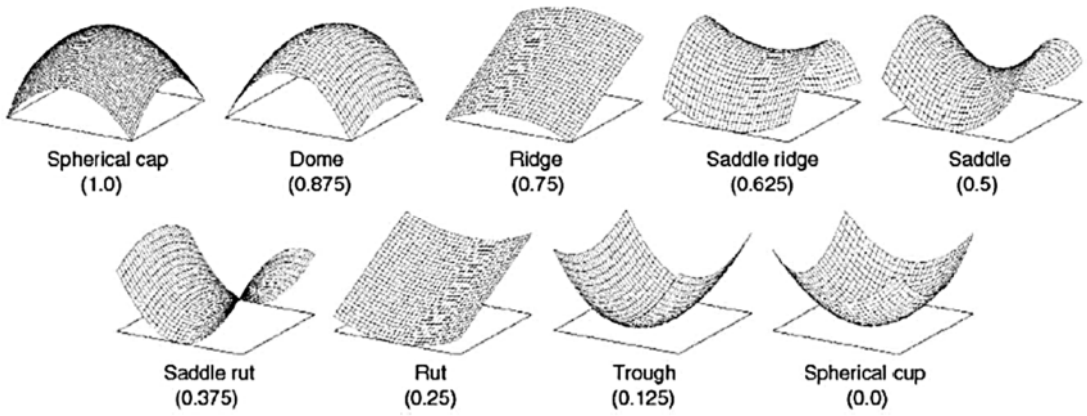
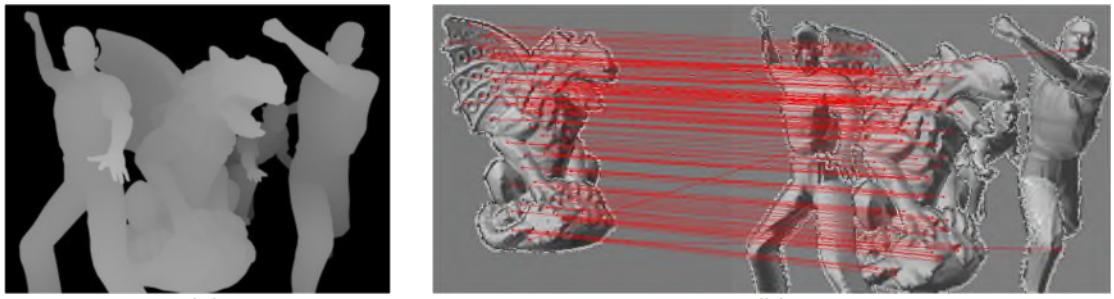


Figure 3- 2 Representations of the Shape Index values (image from [150]**)**

The advantage of *SI* compared to raw depth data is that it emphasises even minor surface anomalies. In addition, the information content per pixel in the *SI* form is more complete because it is affected by its neighbouring pixels and therefore SI-SIFT is more descriptive compared to directly implementing SIFT on the 2.5D images. Figure 3- 3 depicts an example of the SI-SIFT method.

Although SI-SIFT performs well, in the context of military 3D ATR it has two major drawbacks. First, its out-of-plane rotation invariance is limited to ±30° and second, *SI* conversion and SIFT estimation require substantial processing time.



(a) (b)

Figure 3- 3 SI-SIFT (a) 2.5D image (b) *SI* representation and SI-SIFT matches (images from [53]**)**

Lo and Siebert [146] propose a different variant of the *SI* and SIFT combination named 2.5D-SIFT. They z-normalise the range image to zero mean and $\sigma=1$, transform it to a *SI* and then detect SIFT keypoints. Each detected keypoint *P* is assigned with its position *(i,j)*, the scale *r* in which the keypoint is detected in, the canonical in-plane orientation θ, slant φ and tilt t angle given by:

$$\theta = \tan^{-1}\frac{\partial y}{\partial x} \quad \textbf{(3- 6)}$$

$$\varphi = \tan^{-1}\left(\frac{\sqrt{N_x^2 + N_y^2}}{N_z}\right) \quad \textbf{(3- 7)}$$

$$\tau = \tan^{-1}\left(\frac{N_x}{N_y}\right) \quad \textbf{(3- 8)}$$

$$[N_x, N_y, N_z] = \frac{[-f_x, -f_y, 1]}{\sqrt{1 + f_x^2 + f_y^2}} \quad \textbf{(3- 9)}$$

where f_x, f_y denote the first order Gaussian derivatives of the *SI* 2.5D image at coordinates *(i,j)*.

Then on a circular patch of radius *r* centred at *P*, nine elliptical Gaussian weighted regions are placed to obtain the local depth and orientation distribution in the form of a histogram. Depth distribution is based on the typical *SI* shapes presented in Figure 3- 2 after being normalized with the Gaussian curvedness:

$$curvedness = \sqrt{2H^2 - K} \quad \textbf{(3- 10)}$$

Finally, the proposed descriptor is the concatenation of the surface and orientation histograms per ellipse which are normalised to unity. Feature matching is performed through the Nearest Neighbour Distance Ratio (NNDR) [151] technique and matches are verified via a modified Hough Transform scheme. Examples of the 2.5D-SIFT proposal are presented in Figure 3- 4.

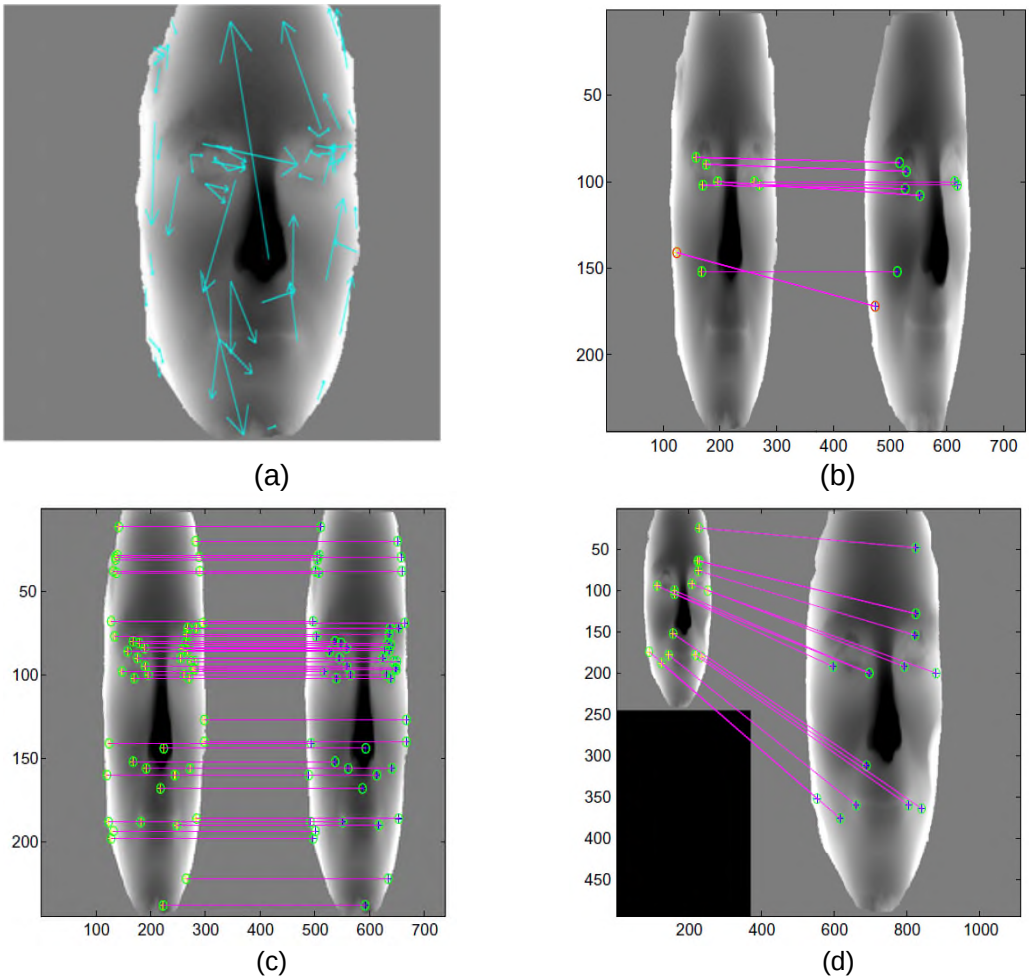


Figure 3- 4 2.5D SIFT (a) 2.5D image showing the 2.5D SIFT features, matching examples in (b) fixed size with 20˚ out-of-plane rotation (c) same scale (d) different scale and (images from [146])

Major drawbacks of the 2.5D-SIFT are the computational burden that *SI* and SIFT features imply [53]. In addition, out-of-plane rotational invariance is constrained to ±30° due to SIFT's limitation and therefore several templates per target are required to accommodate a full 3D rotational invariance.

3.1.1.2 SURF based proposals

Alternative proposals are provided by Krizaj *et al.* [148] and by Lei *et al.* [41] that suggest algorithms based on the processing efficient SURF. The former proposal suggests simply applying SURF to a 2.5D image that is previously transformed to a 2.5D Shape Index image. The latter relies on more complex concepts and converts the raw 2.5D image into a multi-level B-spline approximation. An example is presented in Figure 3- 5.

Even though SURF is approximately five times faster than SIFT [148], estimating the *SI* or converting the 2.5D image into a B-spline are time consuming tasks that exceed the constraints of military applications (a detailed processing analysis of the *SI* estimation is presented in 2.4.2.4). In addition, a single template cannot accommodate a full 3D rotational invariance and therefore several templates per target under various poses are required. This extended template size requirement increases even further the total processing time of the ATR pipeline.

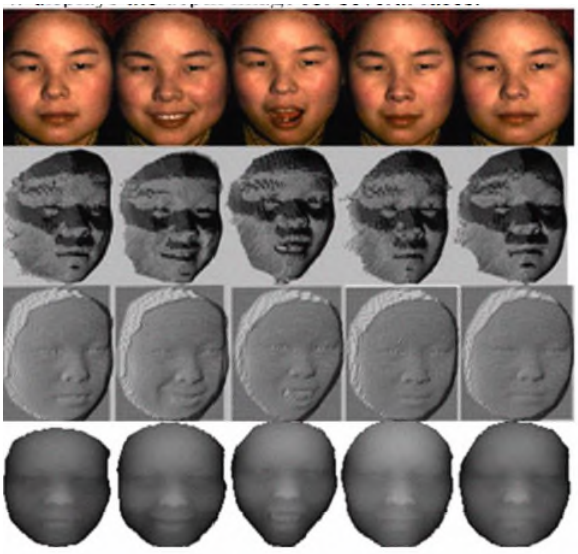


Figure 3- 5 SURF based (a) 2D RGB image (b) 3D point cloud (c) B-Spline resampled model (d) 2.5D image (images from [41]**)**

3.1.1.3 2D binary based proposals

In [148] Krizaj *et al.* challenge SIFT, SURF and a number of 2D binary descriptors i.e. BRIEF [94], ORB [95], BRISK [96] and FREAK [97] on various 2.5D representations in the context of face recognition. For that task the binary descriptors are combined with a keypoint detector, with the detector – descriptor pairs being, FAST [152] - BRIEF, FAST – ORB, AGAST [153] – BRISK and AGAST - FREAK.

The 2.5D representations evaluated are raw 2.5D data, maximum curvature, mean curvature, z-components of the surface normal and the *SI*. As shown in Figure 3- 6, *SI* affords most keypoints and therefore in this paper face recognition trials are based on 2.5D *SI* images.

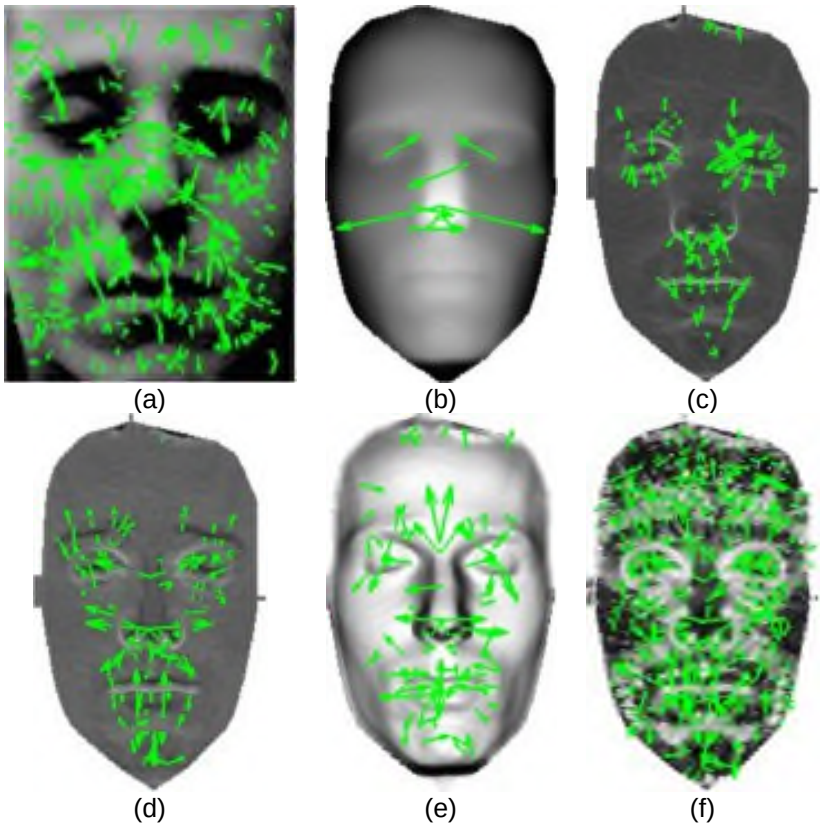


Figure 3- 6 SIFT keypoints detected on (a) grayscale (b) 2.5D image (c) maximum curvature (d) mean curvature (e) z component of the surface normal (f) *SI* (image from [148]**)**

Performance is presented in terms of recognition and of processing efficiency. Although SIFT has a superior recognition performance, the binary descriptors are performing sufficiently well. When it comes to processing efficiency though, the binary ones are substantially faster (Figure 3- 7).

Although this survey highlights the advantages of applying binary descriptors on 2.5D SI images, this strategy has a few drawbacks. First, the computational time in [148], presented in Figure 3- 7, does not include the processing time required to convert the raw 2.5D image into the 2.5D *SI* equivalent. Therefore, the true total processing time is substantially larger. Second, trials do not include rigid transformation and importantly in and out-of-plane rotation. It is safe though to claim that the rotation invariance of each descriptor will be at most the RGB image equivalent, and therefore numerous templates are required to accommodate a full 3D rotation invariance. In the context of missile ATR applications, this will increase the template matching time and thus the computational efficiency of the entire ATR process.

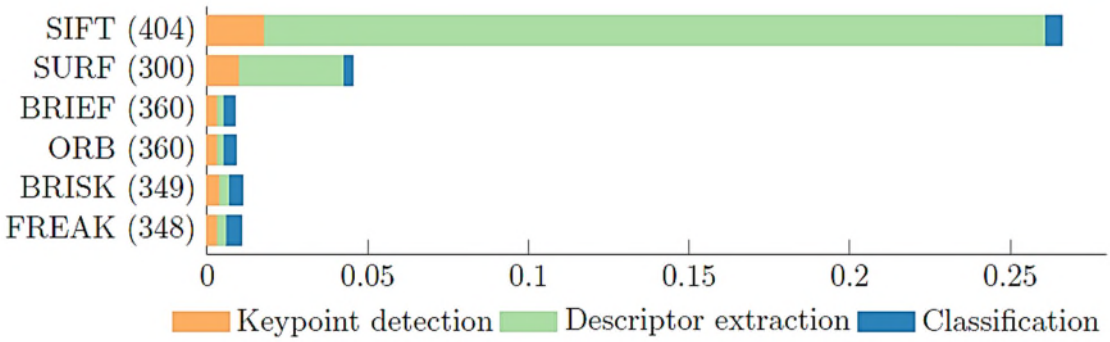


Figure 3- 7 Average processing time in seconds (number of detected keypoints in brackets) (image from [148]**).**

3.1.2 Local Surface patches

Chen and Bhanu in [40], [43] introduce LSP which is a local surface descriptor that is specifically designed for 2.5D SI imagery. The vertices $P_i, \{i \mid i \in \mathbb{N}\}$ that are in the vicinity of a keypoint *P* belong to the latter's support region ***N*** if they

satisfy the constraint of Equation 3-11.

$$P_i \in N \; s.t. \; \left\{ \left\| P_i - P \right\| \le \varepsilon_1 \; \wedge \; a\cos(n_P \cdot n_{P_i}) < A \right\} \qquad \textbf{(3- 11)}$$

where $\cdot$ denotes the dot product between the normal vectors n_P and n_{P_i} at P and P_i respectively. Parameters ε_1 and A are user defined thresholds and determine the descriptiveness of the LSP. Finally, the LSP descriptor at P consists of the keypoint coordinates, the surface type of the support region based on the typical *SI* shapes (Figure 3- 2) and a 2D histogram that encapsulates the relationship between the *SI* of each point $\boldsymbol{N}$ in correlation to the $\langle n_p, n_N \rangle$ angle. The LSP description concept is presented in Figure 3- 8.

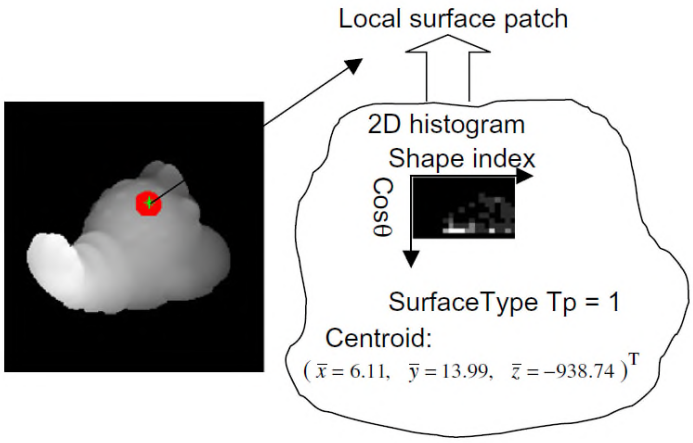


Figure 3- 8 LSP descriptor comprising of a 2D histogram of *SI* vs. angular variation, *SI* based surface type and keypoint coordinates (image from [40]**)**

The main advantage of LSP is its robustness to clutter [90], while drawbacks are: first, the extra processing time to calculate the *SI*. Second, the ±35˚ out-of-plane rotational invariance implying numerous templates per target to accommodate a full 3D rotational invariance. Third, LSP is very sensitive to noise and has a moderate robustness to occlusion [90].

3.1.3 Binary Robust Appearance and Normals Descriptor (BRAND)

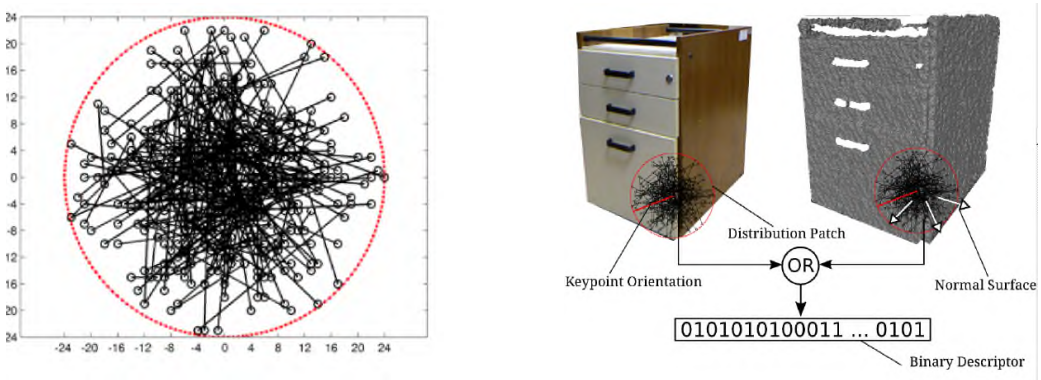
Nascimento *et al.* [100], [101] propose BRAND which is a descriptor specifically

designed for the 2.5D domain. BRAND is a 3D binary descriptor that fuses geometrical cues i.e. depth information and appearance cues i.e. colour texture information. BRAND is unique in exploiting multiple cues and being directly applicable to raw 2.5D imagery.

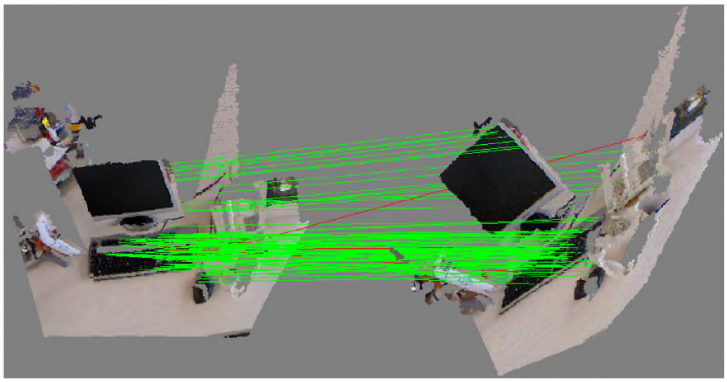
BRAND works as follows: A 48x48-pixel size circular patch is overlaid on the 2.5D image that is centred at a keypoint *P*. For rotational invariance, the dominant orientation of the underlying image is estimated along with the scene's scale. The patch is then aligned according to the dominant orientation and is scale normalized. Then from the same patch, 256 pixel pairs are selected based on an isotropic Gaussian distribution.

BRAND fuses appearance and geometrical information per pixel pairs into a 256-long binary string. The intensity difference of the pixel pairs defines the appearance information. Geometrical information relies on the relationship between the normal displacement and the surface's convexity of each pixel pair. An example of BRAND is presented in Figure 3- 9.

Even though BRAND is computationally efficient and requires a small amount of storage memory, it nevertheless has a few drawbacks. First, it demands colour information that is either not always available or can be prone to illumination variation. Second, full 3D rotational invariance requires a great number of templates to consider all potential viewing poses. Third, BRAND's robustness to noise is questionable because the geometric part of the descriptor is relying on the normal displacement and the surface's convexity that can be influenced by noise.



(a) (b)



(c)

Figure 3- 9 BRAND descriptor (a) Isotropic Gaussian patch for pixel point-pair selection (b) appearance and geometrical data fusion (c) BRAND feature matching example (images from [100]**)**

3.1.4 Discussion on current 2.5D Based 3D descriptors

Even though current 2.5D descriptors have appealing features, overall they do not pose an optimum solution in the context of missile 3D ATR applications because:

a. Most descriptors transform the raw 2.5D image into its *SI* form imposing additional processing time.

b. Principal curvature information used in Shape Index and in BRAND is affected by noise because it involves first and second order derivatives.

c. To achieve a full 3D rotational invariance, a large number of templates is required, such as to consider all possible viewing poses. The number of poses is directly related to the out-of-plane rotation invariance of each descriptor.

3.2 Range Image Based 3D Automatic Target Recognition for Future LIDAR Missiles

Considering those drawbacks, this chapter proposes a near-real time 2.5D ATR algorithm that is suitable for military LIDAR based time-critical applications with limited hardware capabilities. This solution exploits the state-of-the-art 2D local based SURF descriptor which is applied to multiple raw 2.5D projections of a 3D object/ scene. Processing time is further contracted by exploiting the extreme case of a single template per target. The proposed descriptor, named the SURF Projection Recognition (SPR) [20], is invariant to rigid transformations (including scale) combined with Gaussian noise and target subsampling. Applied on military targets from the Princeton shape benchmark and on a set of simulated cluttered and occluded scenarios, more than 90% object class recognition is obtained in less than 100ms for point clouds exceeding 90,000 points. Compared to the state-of-the-art 3D local based descriptor RoPS, it achieves higher recognition rates and one order of magnitude faster execution time and storage memory demand.

3.2.1 The SURF Projection Recognition approach

Given a point cloud $\boldsymbol{P} \in \mathbb{R}^3$, each vertex can be represented as $P_u, \{u \mid u \in \mathbb{N}, u \leq M\}$ where M is the total number of points. Initially P_u is uniformly quantized to P_{qu} with a quantization step Δ to reduce the number of points and thus the overall processing time:

$$P_{qu} = sign(P_u)\Delta \left\lfloor \frac{|P_u|}{\Delta} + \frac{1}{2} \right\rfloor \qquad \textbf{(3- 12)}$$

Each point $P_{qu}, \{qu \mid qu \in \mathbb{N}, qu \leq L, L < M\}$ of the quantized point cloud $\boldsymbol{P_{qu}}$ that contains L points, is then transformed from the missile reference frame *(i,j,k)* to an external Global Reference Frame (GRF) by exploiting information from the missile's gyroscopes i.e. pitch (θ), roll (φ) and yaw (ψ) angles. Both reference frames are centred at the missile seeker, while the *(X,Y,Z)* GRF affords reduced complexity and computational cost. This happens because the *(X,Y,Z)* reference frame does not align with each target in the scene individually, but with the GRF

that includes both the missile and the scene[2]. The coordinates of each point P_{qu} are transformed from the missile reference frame *(i,j,k)* into the GRF *(X,Y,Z)* by applying the Euler – Rodrigues rotation formulas:

$$\begin{aligned} R_\theta &= \cos(\theta)I + \sin(\theta)\lfloor i \rfloor_x + (1-\cos(\theta))i \otimes i \\ R_\varphi &= \cos(\varphi)I + \sin(\varphi)\lfloor j \rfloor_y + (1-\cos(\varphi))j \otimes j \\ R_\psi &= \cos(\psi)I + \sin(\psi)\lfloor k \rfloor_z + (1-\cos(\psi))k \otimes k \end{aligned} \quad \textbf{(3- 13)}$$

for the x-axis and equally for the y-axis and the z-axis:

$$\lfloor i \rfloor_x = \begin{bmatrix} 0 & -k & j \\ k & 0 & -i \\ -j & i & 0 \end{bmatrix} \text{ and } i \otimes i = \begin{bmatrix} i^2 & ij & ik \\ ij & j^2 & jk \\ ik & jk & k^2 \end{bmatrix} \quad \textbf{(3- 14)}$$

where, *I* is the identity matrix, $\lfloor i \rfloor_x$ is the cross-product matrix of *I* and $i \otimes i$ is the tensor product. Transformıng $\boldsymbol{P_{qu}}$ from the missile to the GRF provides a new set of points $\boldsymbol{P'_{qu}}$:

$$P'_{qu} = \begin{bmatrix} x'_{qu} \\ y'_{qu} \\ z'_{qu} \end{bmatrix} = R_\theta R_\varphi R_\psi \begin{bmatrix} x_{qu} \\ y_{qu} \\ z_{qu} \end{bmatrix} \quad \textbf{(3- 15)}$$

where x_{qu}, y_{qu}, z_{qu} are the quantized coordinates in the *(i,j,k)* missile reference frame and $x'_{qu}, y'_{qu}, z'_{qu}$ are the corresponding coordinates in the *(X,Y,Z)* global reference frame.

Projecting each point P'_{qu} to every plane of the GRF is done by the orthographic projection matrix P_{ortho} by zeroing the appropriate binary remapping coefficients $c_1, c_2, c_3 \in \{0,1\}$ from the 3D to the 2D space, depending on the plane on which the cloud will be projected. For example, if $c_1 = c_2 = 0$ and $c_3 = 1$ then the f_{xy} projection plane is received. In parallel, the point cloud is translated to the origin

[2] The GRF used is essentially the typical World Geodetic System 84 (WGS 84) used by GPS systems translated to have an origin at the missile LIDAR seeker

of the global reference frame which is set at the missile's seeker by applying the proper translation coefficients t_1, t_2, t_3.

The coordinates $\widetilde{P}$ of the orthographically projected point cloud after being quantized, rotated to the global reference frame and translated to the origin, are given by:

$$\widetilde{P} = \begin{bmatrix} \tilde{x}_{qu} \\ \tilde{y}_{qu} \\ \tilde{z}_{qu} \\ 1 \end{bmatrix} = P_{ortho} P'_{qu} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x'_{qu} \\ y'_{qu} \\ z'_{qu} \\ 1 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ 1 \end{bmatrix} \quad \textbf{(3- 16)}$$

where $\tilde{x}_{qu}, \tilde{y}_{qu}, \tilde{z}_{qu}$ are the coordinates of the orthographically projected points on the f_{XY}, f_{XZ}, f_{YZ} plane in respect. The three orthographic projections are 2.5D range images i.e. simplified versions of the 3D point cloud $\boldsymbol{P'_{qu}}$. In these images, the depth value of each plane i.e. $f_{XY} = (\tilde{x}_{qu}, \tilde{y}_{qu}) = \tilde{z}_{qu}$ is unique and represents the distance between the target and the LIDAR seeker. Figure 3- 10 presents an illustration of the reference frame conversion and the 2.5D projections.

The size of each projection is variable depending on the amplitude of the point cloud values after quantisation. During the final pre-processing step, before the SURF keypoint detection and description stage, the range images are rescaled into a fixed size of 128pixels*W, where W is the width of the projection, with W≥128. This strategy maintains the aspect ratio [154] and avoids image distortion. In parallel, the fixed sized projections aim at further reducing the processing time and improving the ATR performance over a greater range of scales.

Although the quantisation process reduces processing time, it inevitably imposes information loss that can downgrade the recognition quality. Thus, a balance between recognition performance and processing time is crucial.

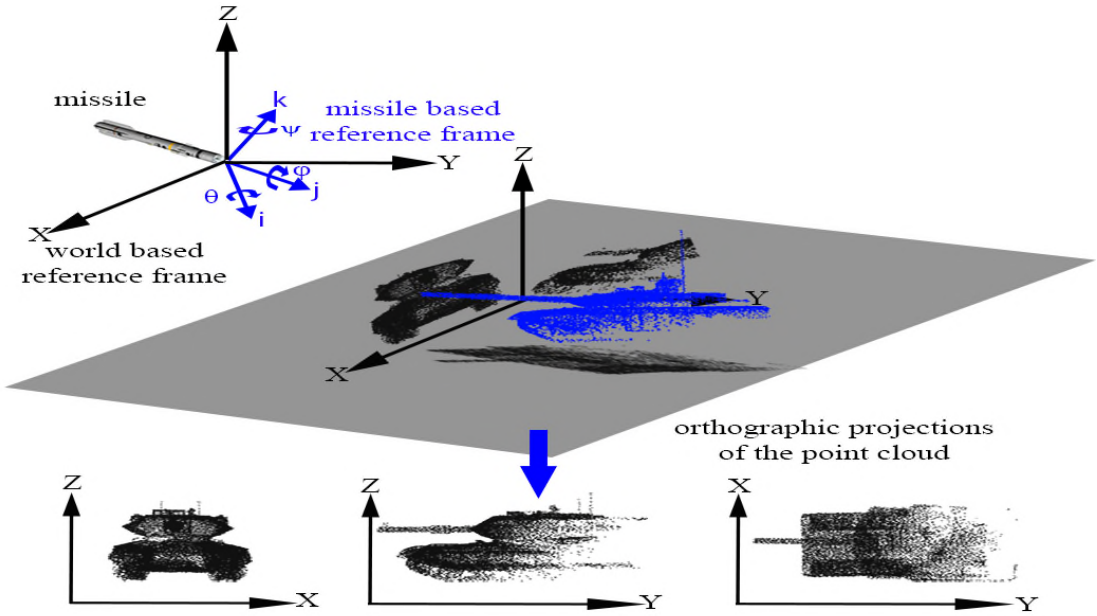


Figure 3- 10 3D to multiple 2.5D transformation. M1A1 Abrams MBT (blue) as observed from the missile's reference frame (blue). The MBT is quantized and transformed to the GRF (black) after incorporating information from the missile's gyroscopes. Range images are created from the projection of the MBT onto the planes of the (X,Y,Z) GRF coordinate system (image from [20]**)**

3.2.2 Local Features

In [93] Bay *et al.* proposed the Speeded-Up Robust Features (SURF) as a faster counterpart of the popular SIFT [92]. SURF is a stand-alone solution as it contains a 2D keypoint detector and a descriptor. Initially, SURF creates a response map and detects points of interest based on the local extreme of the approximated determinant of the Hessian (H_{approx}) :

$$\arg local\max\left(Det(H_{approx})\right) = \arg local\max\left(D_{xx} - D_{yy} - (0.9D_{xy})^2\right) \qquad \textbf{(3- 17)}$$

where D_{xx}, D_{yy}, D_{xy} are the discretized versions of the corresponding Gaussian second order kernel convolved with the 2D projection of interest:

$$D_{xx}(\tilde{x}_{qu},\tilde{y}_{qu},\sigma) = \frac{\partial^2}{\partial \tilde{x}_{qu}^{\,2}}(sign(g(\sigma))\Delta\left\lfloor \frac{|g(\sigma)|}{\Delta} + \frac{1}{2}\right\rfloor)\cdot f(\tilde{x}_{qu},\tilde{y}_{qu}) \qquad \textbf{(3- 18)}$$

where *f* is the 2.5D orthographically projected plane of the GRF, *g* is the Gaussian kernel of standard deviation σ and Δ the quantization step.

During the keypoint detection phase of SPR on each of the three projected 2.5D images, SURF relies on three octaves and four scale intervals per octave. The threshold of the approximated determinant of the Hessian is set to 10^{-5}. SPR uses the default 64-element long SURF descriptor.

The quantization step Δ applied to the initial point cloud is crucial as it affects the number of detected keypoints and therefore the overall performance. Specifically, as the quantization step Δ decreases, SURF detects more keypoints because finer details of the scene are revealed (Figure 3- 11). In contrast to the B-spline [41], this pre-processing step has minimum computational cost.

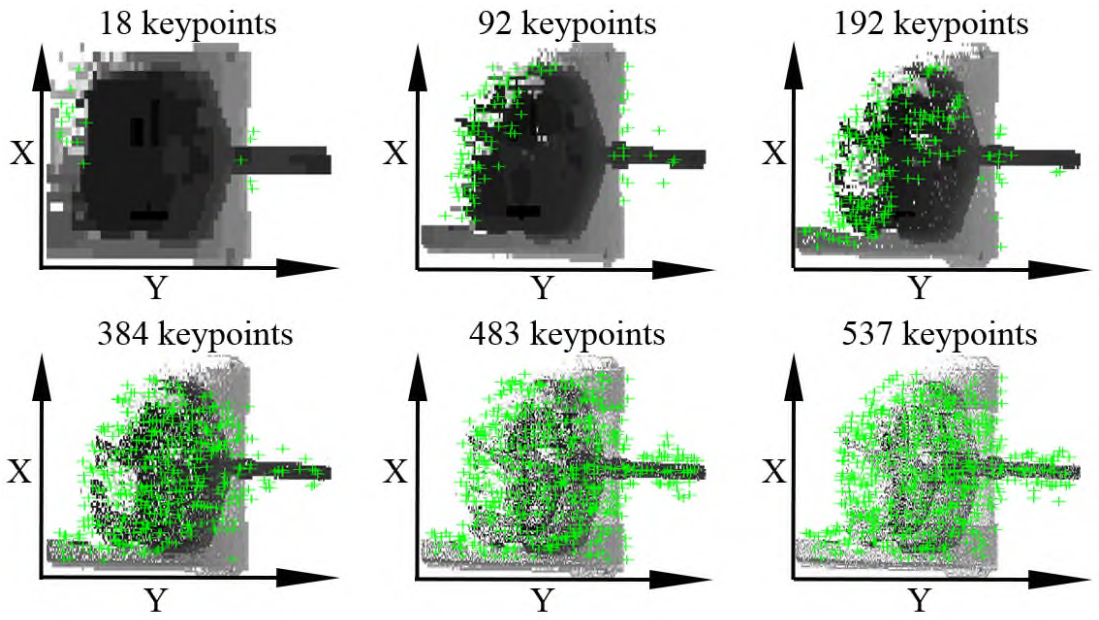


Figure 3- 11 Top view projection of the M1A1 MBT with the FAST Hessian keypoints shown at different quantization steps (image from [20]**)**

Given a set of model features f_i^m, a ground truth transformation and the corresponding scene features f_i^s a scene feature is matched with all model features based on a Euclidean sum of squared differences metric and the NNDR [92] criterion. For the NNDR, if the ratio of the nearest model feature f_i^m with the second nearest f_i^m is less than a threshold τ set at 0.6, then the scene feature f_i^s

and the model feature f_i^m are considered as a match.

As reported by the developer of SURF, the latter has a stable performance in the scale range from one up to 2.5. However, beyond that region performance dramatically decreases. ATR algorithms that should exceed this restriction include extra training sets with the expected target in various scales. In this case, the size of the database and the matching time significantly increase.

In SPR, the recognition capability over several scales is extended by resizing both the template's and the target's range images to a fixed size of 128pixels*W, where W≥128. The aspect ratio is preserved to avoid image distortion while the resizing procedure is approximated by nearest-neighbour interpolation for computational efficiency. In addition, the database includes a set of potential target templates using small sized range images, simulating the target being at the furthest range or in equivalence, in the smaller scale that the sensor can detect. This methodology provides various advantages:

a. Scale invariance can exceed SURF's constraint without increasing the size of the database.

b. As the missile moves towards the target, the size of the target increases directly influencing the number of the detected keypoints and substantially increasing the processing time to detect, extract and match the features. In SPR, resizing the range images to a small and fixed size, regardless of the true size, provides a predictable number of keypoints. This is achieved with a minor computational expense since the efficient nearest-neighbour interpolation is used.

c. SURF achieves most matches when both the target and the template are in the same scale. By resizing, as in this approach, the target's 2.5D image to a fixed size, the number of matches is maximized maintaining a high and quite stable recognition performance.

d. Additionally, as the missile – target range reduces, each 2.5D image of the target is downscaled creating a smoothed version that discards some of

its details. The smoothed images compensate a robust recognition performance even under noisy or sparse target data.

3.2.3 Hough Pose Filtering

Even after matching the SURF features, outliers may still exist that can be discarded by applying Hough Pose clustering [92]. This filtering method is based on a voting process where the already matched keypoints from SURF are re-matched in a Hough space over scale σ and rotation θ [155].

For the SPR in specific, the matched keypoints of the scene f_i^s and template model f_i^m are plotted on a 2D accumulator plane where the x-axis represents the scale bins σ and the y-axis the orientation bins θ in which the matched keypoints are detected. An accumulator plane is a plane where each keypoint occupies a bin based on its σ and θ combination where it is detected. So, each matched keypoint from the NNDR stage votes for a single bin in the accumulator plane of the target and the template respectively. Finally, a cluster of refined matches is created as the intersecting bins of both accumulator planes. In case more than one matched pair of keypoints occupies the same bin, only the first pair is considered as being valid. To reduce discretization errors, the scale bins have a size of one and a range from one up to 20 and the rotation bins have an increment of 15° in the 0°-360° range.

Figure 3- 12 presents an example where the NNDR threshold provides 76 matches between two different MBT types. Each matched pair, depending on the scale σ and orientation θ occupies a single bin in the template and the target accumulator plane in respect of the Hough space. The intersection of both accumulator planes creates clusters that provide a refined set of matched keypoints reducing mismatches by 91%.

Feature matching refinement is also possible based on methods exploiting geometric constraints such as the RANSAC [156] or by extending the Correspondence Grouping concept from the 3D data domain to the 2D [40], [43]. The downside of both these solution though is their iterative nature that inevitably

increases the overall processing time.

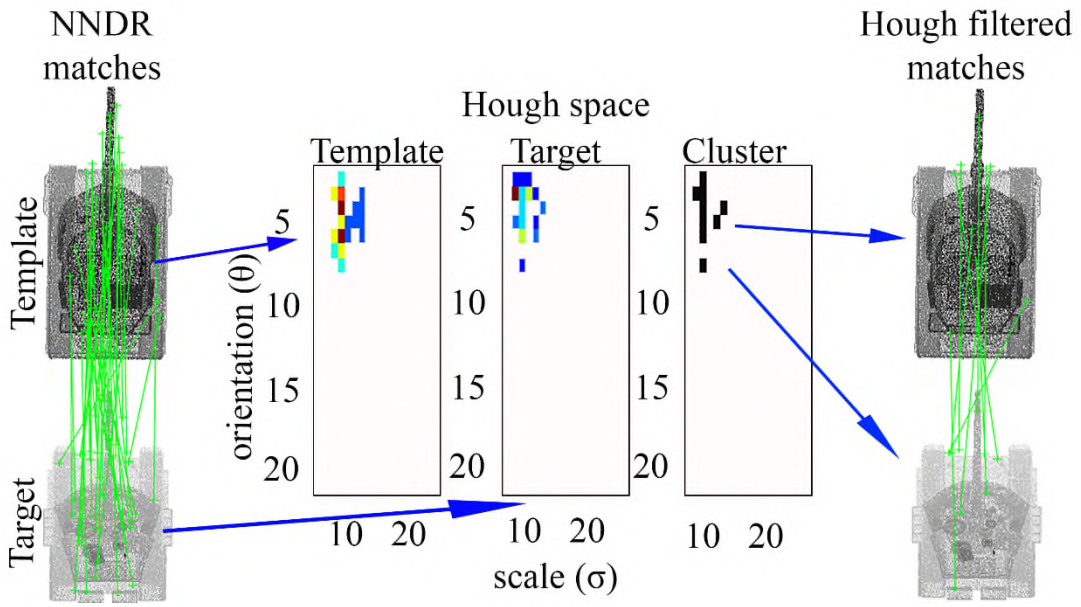


Figure 3- 12 Hough pose filtering. NNDR matches are re-matched in the Hough space and fill the accumulator plane of the target and the template. Common scale and orientation bins of both accumulator planes create clusters of refined matches. The bin color represents the number of accumulated matches for that σ and θ combination (image from [20]**)**

3.2.4 Simulating viewing dependent point clouds

Most available models are in a 3D ideal representation while in reality the LIDAR seeker can only receive a part of the target depending on the orientation of the target to the LIDAR device. Typical missiles against ground targets rely on top and side view attack in order to defeat the target where armour is thinnest. Thus, the Hidden Point Removal (HPR) [157] algorithm is used to create self-occluded point cloud views emulating realistic views of the LIDAR missile seeker. HPR includes three stages. Initially, it remaps the coordinates of each point P_m of the raw point cloud to a mirror image as observed from the viewpoint. This is done by using an imaginary ray connecting each point P_m and the viewpoint, which is set at the missile's LIDAR seeker. The next step incorporates the projection of the remapped point cloud onto a sphere of radius R centred at the missile seeker. This procedure is named *spherical flipping* and the resulting point cloud consists

of the P_{sfm} points:

$$P_{sfm} = P_m + 2(R - \|P_m\|)\frac{P_m}{\|P_m\|} \quad \textbf{(3- 19)}$$

For SPR, the radius R is automatically calculated as suggested by Alsadik, Gerke and Vosselman [44]. Finally, the convex hull of the resulting point cloud is given by:

$$\left\{\sum_{m=1}^{c} a_m P_{sfm} \mid (\forall m : a_m \geq 0) \wedge \sum_{m=1}^{c} a_m = 1\right\} \quad \textbf{(3- 20)}$$

where c is the cardinality of the P_{sfm} and a_m a weight factor that is automatically selected by the convex hull algorithm.

Summarizing, a point P_m of the raw point cloud is considered as visible, only if its spherical flipped form P_{sfm} is on the convex hull. The HPR concept is shown in Figure 3- 13.

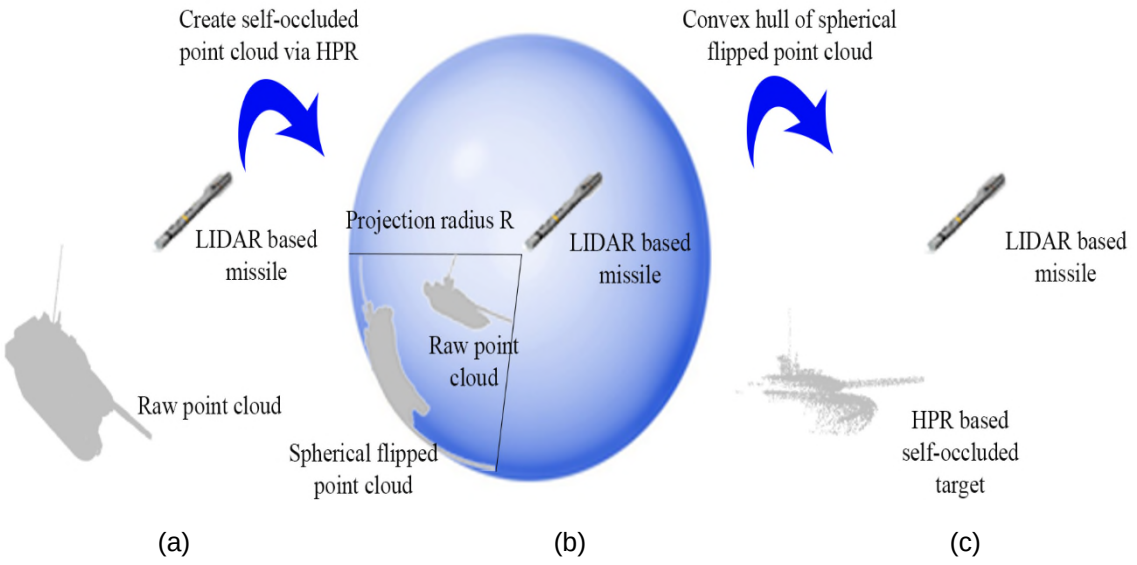


Figure 3- 13 HPR concept (a) LIDAR missile looking at a MBT (b) the raw point cloud of the MBT is flipped and then projected onto a sphere of radius *R*. (c) only points belonging on the convex hull of the spherical flipped point cloud are considered as points of the self-occluded target (image from [20]**)**

3.2.5 SPR based 3D ATR workflow

The SPR procedure can be split into an offline and an online part. Offline, a database of potential targets is created. The ideal 3D point cloud of each target is quantized and orthographically projected on the f_{xy}, f_{xz}, f_{yx} planes of the *(X,Y,Z)* GRF system that are resized to a fixed size. SURF is then applied on the range images created. Each target is represented by three 2.5D range images which are encoded by SURF. The SPR technique exploits the SURF keypoint coordinates, scale σ, orientation θ, and the SURF descriptor. During this stage, it is important to orient each template in its canonical pose and create the 2.5D projections from 45° viewing angle in any axis.

The online description procedure is the same as the offline, except that HPR is applied to simulate the self-occlusion effect. The extracted SURF features are then matched via an NNDR criterion and the template that receives the most matches over the three planes is considered as the recognized target. The NNDR criterion is set to 0.6 such as to balance recognition performance and robustness to perturbations like noise and sparsity. Matches are then refined via the Hough pose filtering scheme presented in Section 3.2.3. In case more than one template provides the same number of maximum matches, target recognition is based on a matching quality criterion. The latter defines the matching quality per target – template match based on the average difference of the responses of the matched SURF keypoints as given from the approximated determinant of the Hessian. The template providing the smallest difference to the target over the three planes is chosen as the recognized one:

$$\underset{template}{\arg\min}\left(\frac{1}{3}\sum_{projection}\left(\overline{H^{m}_{approx}}-\overline{H^{S}_{approx}}\right)\right) \qquad \textbf{(3- 21)}$$

The processing flow of SPR is presented in Figure 3- 14. Figure 3- 15 presents a SPR matching example. It shows the case where a MBT target is rotated 60° in pitch, roll and yaw, self-occluded and at scale *x2s* and is matched to the same and a different MBT class, which are in their canonical pose, without any occlusion and at scale *s*. Each target and template are orthogonally projected to the planes of the GRF to create three distinct 2.5D images. SPR successfully

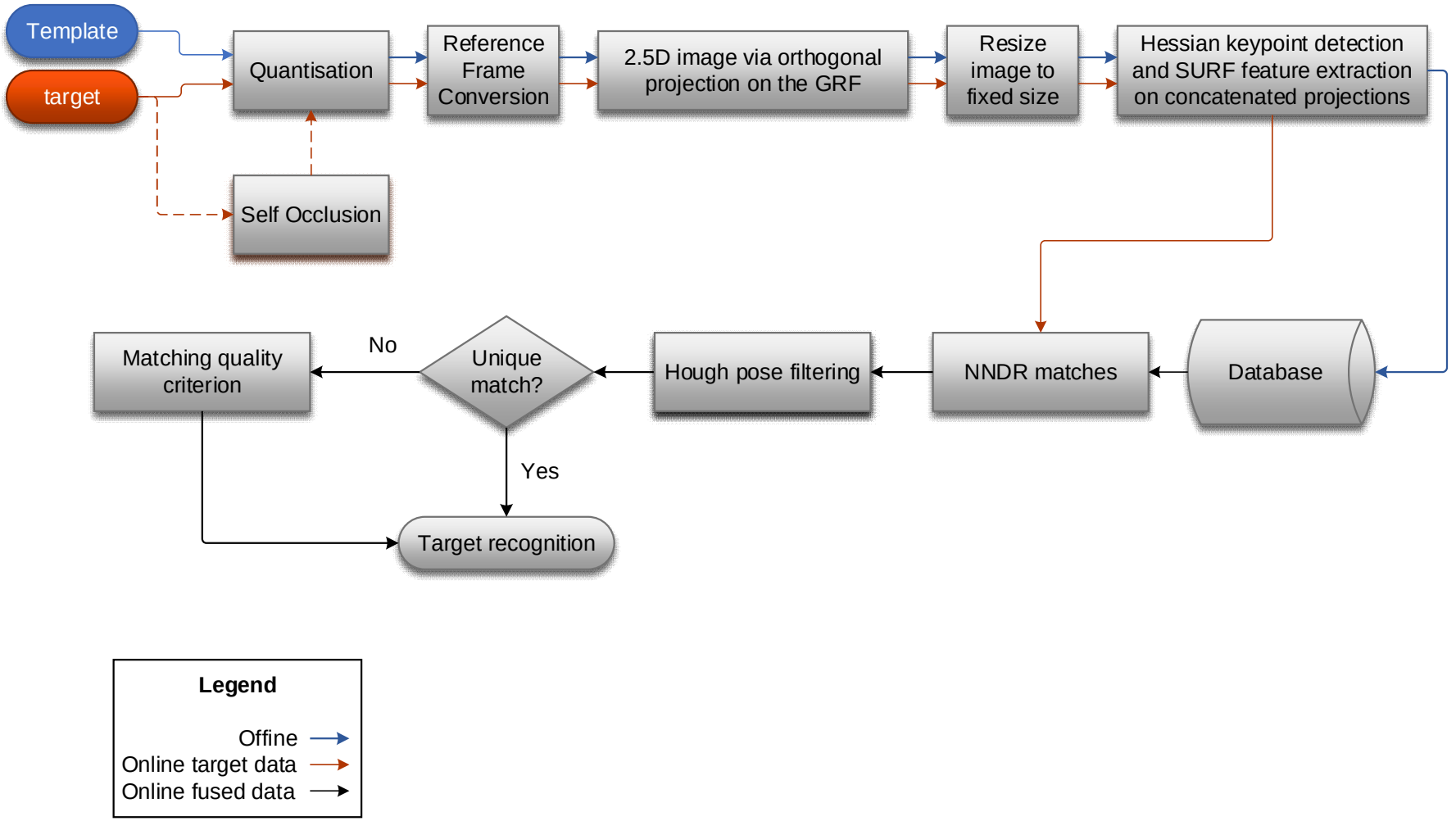
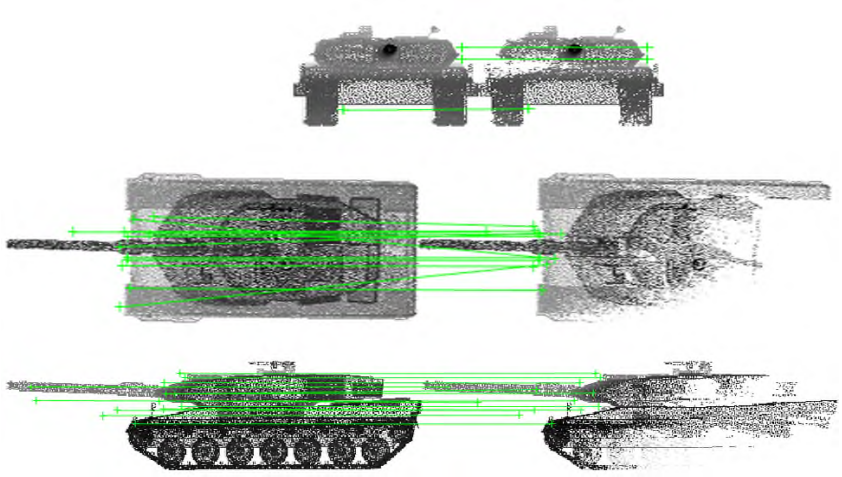
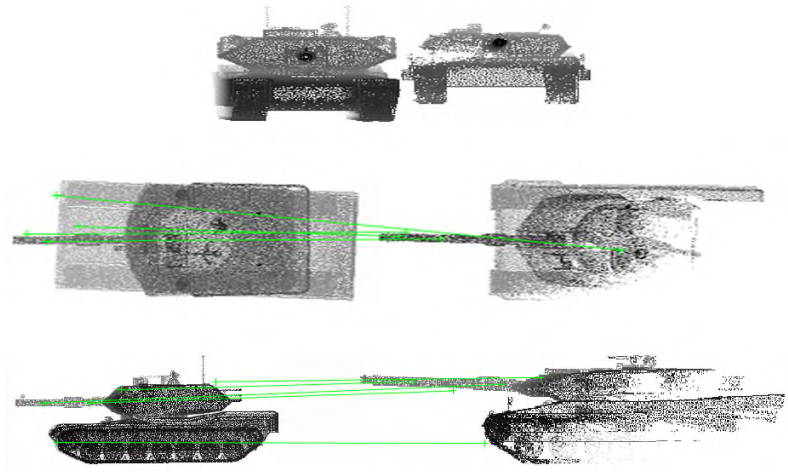


Figure 3- 14 Flow chart of the SPR target recognition algorithm. The self-occlusion process is optional depending on the nature of the scene (real or synthetic) (image from [20]**)**

matches the target with its corresponding template providing in total 28 matches over the three projection planes. On the contrary, for the different MBT class SPR provides only 9 matches. These mismatches mostly occur at the barrel of the MBT as both templates possess one. The availability of more detailed target set data, which gives turret shape or road wheel configuration, would assist in further enhancing discrimination among MBT classes.



(a)



(b)

Figure 3- 15 SPR matched keypoints between (a) same MBT classes - 28 matches and (b) different MBT classes - 9 matches (f_{xy}, f_{xz}, f_{yx} planes from top to bottom). For each plane, left point cloud represents the template and right the target (image from [20]**)**

3.2.6 Experiments

The effectiveness of SPR is challenged on military targets of the Princeton Shape Benchmark database [158] and on ground surface military targets [159] that have inter and intra-class variation. Inter-class variation refers to recognising different classes of targets e.g. a fighter aircraft from a warship. Intra-class variation denotes recognizing different types of the same class, e.g. a M1A1 MBT from a T-90 MBT. In the following experiments, each target is rotated in pitch, roll and yaw in the 0° - 360° region with an increment of 30° neglecting non-applicable poses i.e. all bottom-up viewing variants. The rationale behind the 30° rotation increment is due to the limit of the affine transformation that SURF can manage [93].

Experiments comprise of various combined rigid transformations and perturbations such as Gaussian noise and uniform sparse representation of the target. Trials include target-template scale changes varying from *s* up to *10s*. Initial experiments assume uncluttered targets, while more complicated scenarios are examined in the following sections. All experiments consider the non-target recognition case and self-occlusion.

According to open source data, the processing power of a missile is in the order of a Quad Core PowerPC G4 from the 74xx processor family and ATR algorithms for missiles are implemented in C/C++ [160]. SPR is developed in MATLAB and the processing platform for all trials is an AMD Dual Core 2.1 GHz laptop exploiting a single core. Although this developing scheme differs in relation to a final missile implementation affecting the measured processing time during trials, it is considered that SPR still meets the time response criteria. Specifically, the efficiency of C/C++ compared to MATLAB is in the range of x9 – x500 [161], [162] and the processing efficiency of a missile processor compared to the CPU used is x2.5 [163]. Thus, the overall processing gain of a final missile implementation is x22 up to x1250. That gain increases even more if ordinary processors are substituted by Field-Programmable Gate Arrays (FPGA).

According to future upgrades to the US Navy SM-3 missile, proposed by the MIT Lincoln Laboratory [160], the desired missile latency should be 16.7ms which is

adopted in this research. Considering the processing gain due to the CPU used during trials, the coding differences and this latency, an upper processing time limit of 500ms is set. Although these processing differences are only assumptions while the final speedup has to be verified experimentally, this research is considered more as a feasibility study of rather than a ready-to-use solution. Literature suggests measuring computational complexity in seconds [9]–[11] but due to the processing time limit set and the high-speed the missile is flying at, computational complexity is calculated on a millisecond basis [13].

3.2.6.1 Princeton shape benchmark

One representative of each military target class from the Princeton shape benchmark is used, namely a MBT, a Warship, a Helicopter and a Fighter aircraft as shown in Figure 3- 16.

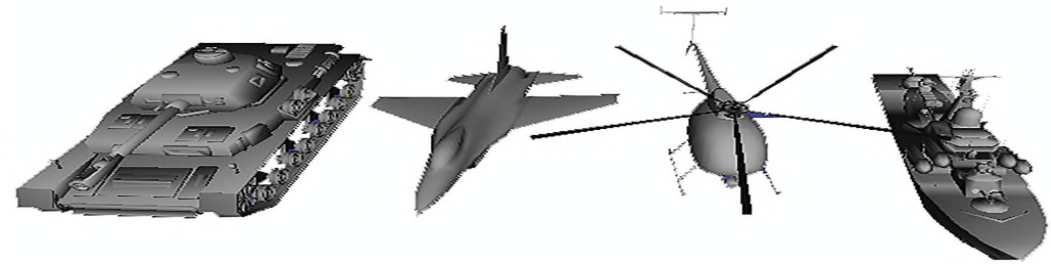


Figure 3- 16 Typical military targets from the Princeton database benchmark (image from [20]**)**

This database has a collection of point clouds generated from CAD models that have a relatively small number of points and with the planar surfaces not fully represented as they have points only at their edges. To provide a realistic representation of those models, points are populated with Poisson sampling [164] increasing their ideal 3D point cloud to 140,000 points per target on average.

In the first set of trials, the missile-target range is the generic *s* while in the second set it is *10s*. Each batch of experiments includes the cases of target 3D rotation, 3D rotation combined with noise, 3D rotation and 50% sparse representation and finally all nuisances simultaneously. During all trials SPR provides high recognition performance with detailed results shown in Figure 3- 17.

In the first experiment, although the target is forced to simultaneous rotation in

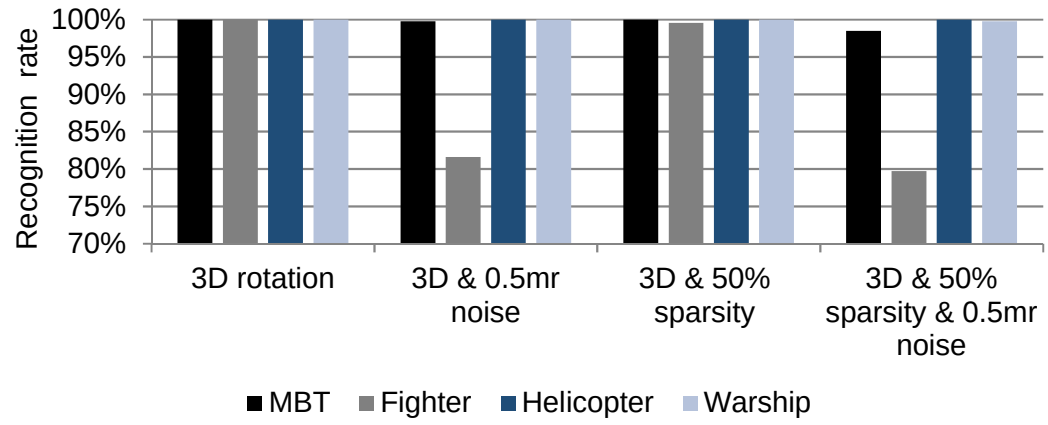
pitch, roll and yaw, SPR manages 100% recognition rate in 238ms. The 3D rotational invariance of SPR is expected due to the complementary nature of the three range images.

The following experiment involves sensor noise and investigates its effect on the recognition performance while the target is forced to 3D rotation. During noise trials, Gaussian noise is considered for the reasons explained in Appendix C. Consequently, Gaussian noise with zero mean and standard deviation equal to 0.5 of the average point cloud resolution (mr) is added to the target. Expressing σ as a multiple of the average mesh or point cloud resolutions is the norm in 3D pattern recognition [60], [63], [89], [112], [165]. The chosen standard deviation is one of the highest values experimented in the current 3D object recognition literature [59], [90]. Although adding noise creates virtual keypoints that can be mismatched, the average recognition capability is still very high at 95.3%. Indeed SPR incorporates SURF and therefore robustness to noise [166] is anticipated which is further enhanced due to the smoothing process introduced in SPR's architecture.

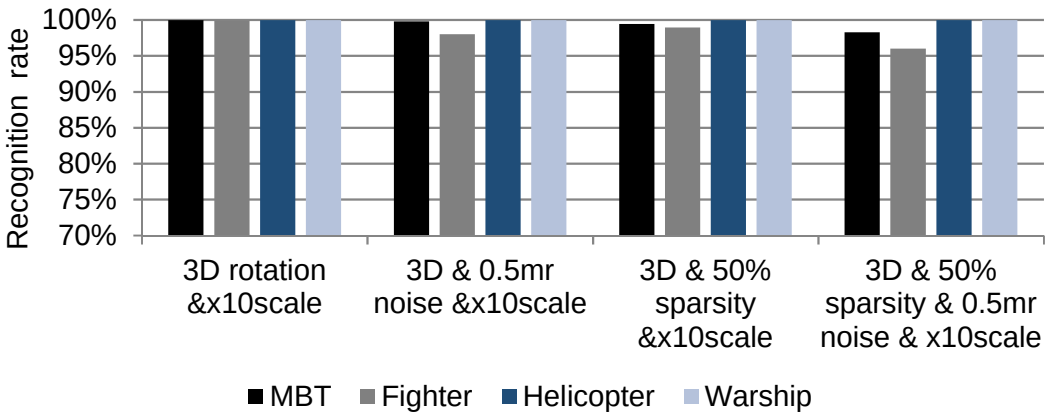
Although SPR achieves a high average recognition rate, the performance of the fighter aircraft is negatively affected. This is because adding noise to the fighter aircraft modifies largely its smooth surfaces, forcing the Hessian keypoint detector to detect false keypoints. So, depending on the viewing angle, these keypoints are falsely matched leading to a performance drop.

Atmospheric conditions may attenuate the laser beam and thus reduce the point cloud density. Therefore, SPR is evaluated in 3D rotation and 50% uniform target sampling. Results show that the overall performance is unaffected achieving 99.9% recognition. This can be explained by the fact that when resizing each 2.5D image projection, it becomes smoother compensating sparse data. Finally, in the last trial that has simultaneous 3D rotation, 0.5mr Gaussian noise and 50%-point cloud decimation, SPR still provides 94.5% recognition rate. Incorporating noise to the targets modifies the flat surfaces of the fighter, reducing its recognition rate in the same manner as in the pure noise case. Detailed results are presented in Figure 3-17 (a).

The same set of trials is executed with the target at scale *x10s*. Increasing the target's scale does not affect the recognition rate of SPR (Figure 3-17 (b)). As expected, the influence of noise is now eliminated through the resizing procedure of the three projection planes. Therefore, the fighter's recognition performance is only minor affected by noise.



(a)



(b)

Figure 3- 17 Performance of SPR on the Princeton database benchmark at scale (a) *s* (b) 10*s*

Evaluating SPR on a military subset of the Princeton shape benchmark reveals the high robustness of SPR to target class recognition under 3D rotation combined with noise, uniform sparse representation and scale change. The next

dataset challenges the proposed technique with targets having both inter and intra-class variation.

3.2.6.2 Surface target CAD model database

A database fitting the scenarios of the ground target case is created. It consists of a missile battery, a Leopard 2A6 MBT (GER), an M1A1 Abrams MBT (USA), a T-90 MBT (RUS) and the auxiliary vehicle Raba H25 shown Figure 3- 18. Each 3D ideal target consists of 115,000 points on average after being populated with Poisson sampling. This database is more challenging compared to the previous one since it comprises of three similar 3rd generation MBTs while at the same time the anti-air missile battery has the body of a MBT. As previously done, all experiments consider the non-target recognition and self-occlusion via HPR is considered.

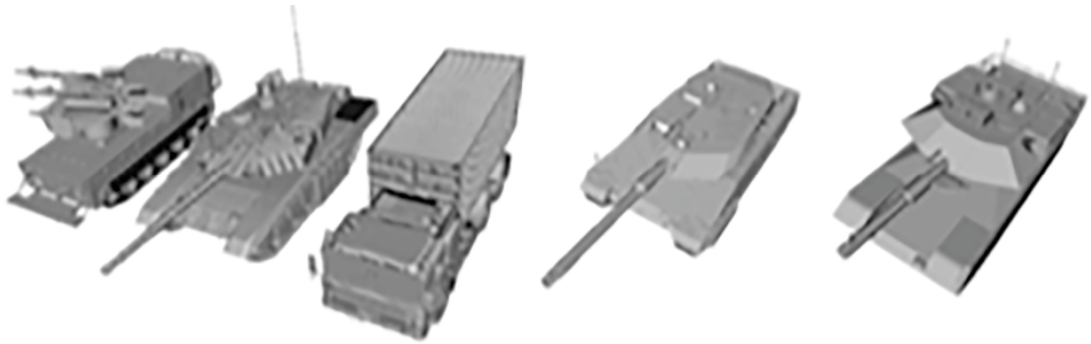


Figure 3- 18 Ground target set: missile battery, T90 MBT, Raba H25, M1A1 Abrams MBT and Leopard 2A6 MBT (image from [20]**)**

Overall, SPR maintains its high recognition performance during all trials with detailed results presented in Figure 3- 19 (a) for the scale *s* case. At scale *s*, with self-occlusion, SPR manages for the 3D rotation case 99.8% in 469ms. Compared to the Princeton shape benchmark trials, processing time has increased because this database is larger and has more complex targets that provide more keypoints that must be matched.

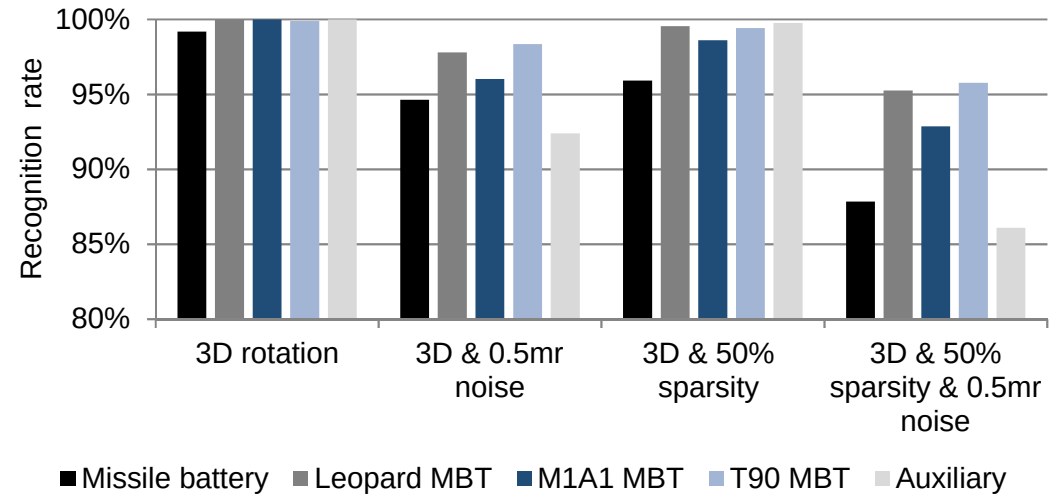
In the next experiment, SPR is evaluated against simultaneous target 3D rotations and 0.5mr Gaussian noise. Although targets have a great similarity, SPR correctly recognizes 95.8% of the cases. The largest performance drop is observed for the auxiliary vehicle as noise alters its flat surfaces, creating false

keypoints which lead to mismatches. Although the recognition rate for the auxiliary vehicle is reduced, SPR still achieves 92% for that target which is considered adequate. This effect of noise is similar to the fighter aircraft of the Princeton shape benchmark trial.

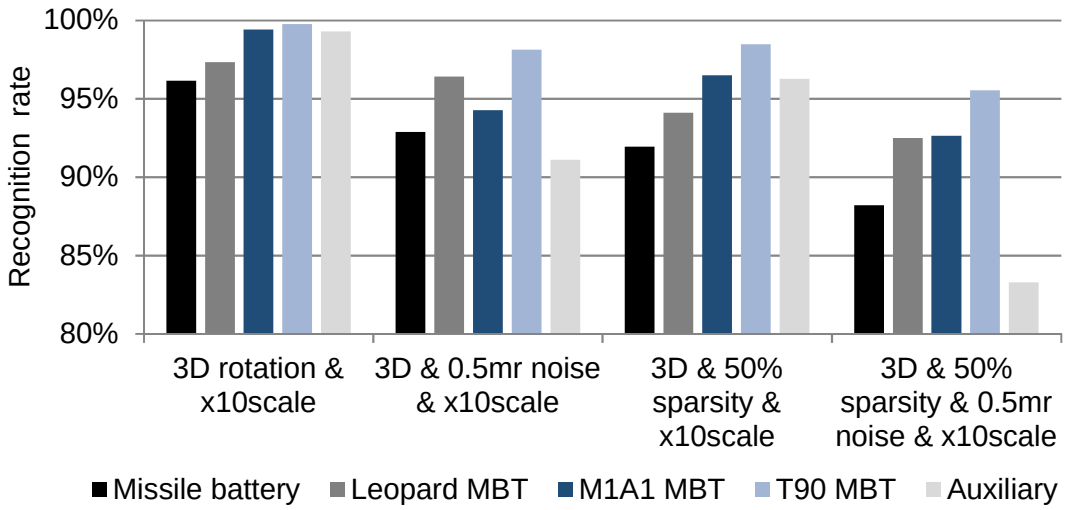
The following trial combines simultaneous 3D rotation and 50% uniform target subsampling. The average recognition rate is 98.6% with the anti-air missile battery having the lowest performance (95.5%), largely because of its main body which is resembles the other MBT type targets.

The next experiment investigates SPR's performance under simultaneous 3D rotation, 0.5mr Gaussian noise and 50%-point cloud decimation. Although this trial combines all perturbations, SPR still achieves high performance managing a 91.6% recognition rate. In this case, although the flat surfaces of the auxiliary vehicle are influenced by noise, recognition is still greater than 85%. Considering that the simultaneous disturbances applied pose a challenging scenario, this performance is still notable.

The second batch of trials evaluates SPR under the same perturbations and transformations with the target at scale *x10s*. The average recognition rate of all trials is now 94.7% in 495ms showing again the strong robustness of SPR under scale change. Similarly to the previous trials, the flat surfaces of the auxiliary vehicle are affected by noise creating false keypoints and influencing recognition. Even in the case where all perturbations and transformations are combined simultaneously, the recognition rate of the auxiliary vehicle is still greater than 83%, which is again considered notable. Detailed performance is shown in Figure 3- 19 (b).



(a)



(b)

Figure 3- 19 Performance of SPR under various trials on the ground surface dataset at scale (a) *s* (b) *10s*

3.2.6.3 Evaluation on military forested scenes

Depth variation due to the relative position of the target within the scene is crucial for the performance of SPR. To compensate that, automatic target detection and then recognition in various forested scenes is performed by rejecting the ground and the tree tops [12].

Three forested scenes with increasing difficulty are evaluated that include multiple targets per scene and clutter objects on a non-planar non-smooth ground surface. Figure 3- 20 presents the scenarios evaluated as observed from the seeker and after being processed to a fixed size of 128pixels*W. In addition, Figure 3- 20 shows the point-to-point matches between the template that provides most matches and the scene.

The first scenario considers the case of a T90 MBT, which is partially occluded by a tree. SPR detects and recognizes the target in 502ms. Specifically, SPR manages to match two out of the three projections of the T90 MBT template. The front side projection has most matches, because after resizing the 2.5D scene projection, the MBT size within the scene is of similar size to the corresponding template projection. The side projection does not provide any matches because the lower part of the MBT in the scene is rejected as ground. This region has distinct features that could provide template-scene matching keypoints. Finally for the top projection, despite tree tops being discarded, the remaining parts of the tree influence the depth values of the MBT and thus the SURF features are limited.

In the second scenario, the scene comprises of a T90 MBT, which is occluded by trees. It is worth noting that the MBT in the scene has a different orientation and scale compared to the template. Still, under these conditions, SPR is able to detect and recognize the MBT in 395ms. Comments per projection of trial one are equally valid for this trial.

In the third scenario, the scene contains two targets, namely an anti-air missile battery and a T90 MBT. Both targets are partially occluded by trees and have a different scale compared to the templates. Positive detection and recognition of both targets is achieved in 307ms. Even though in both cases a small number of mismatches occur, SPR is still capable to provide correct target recognition.

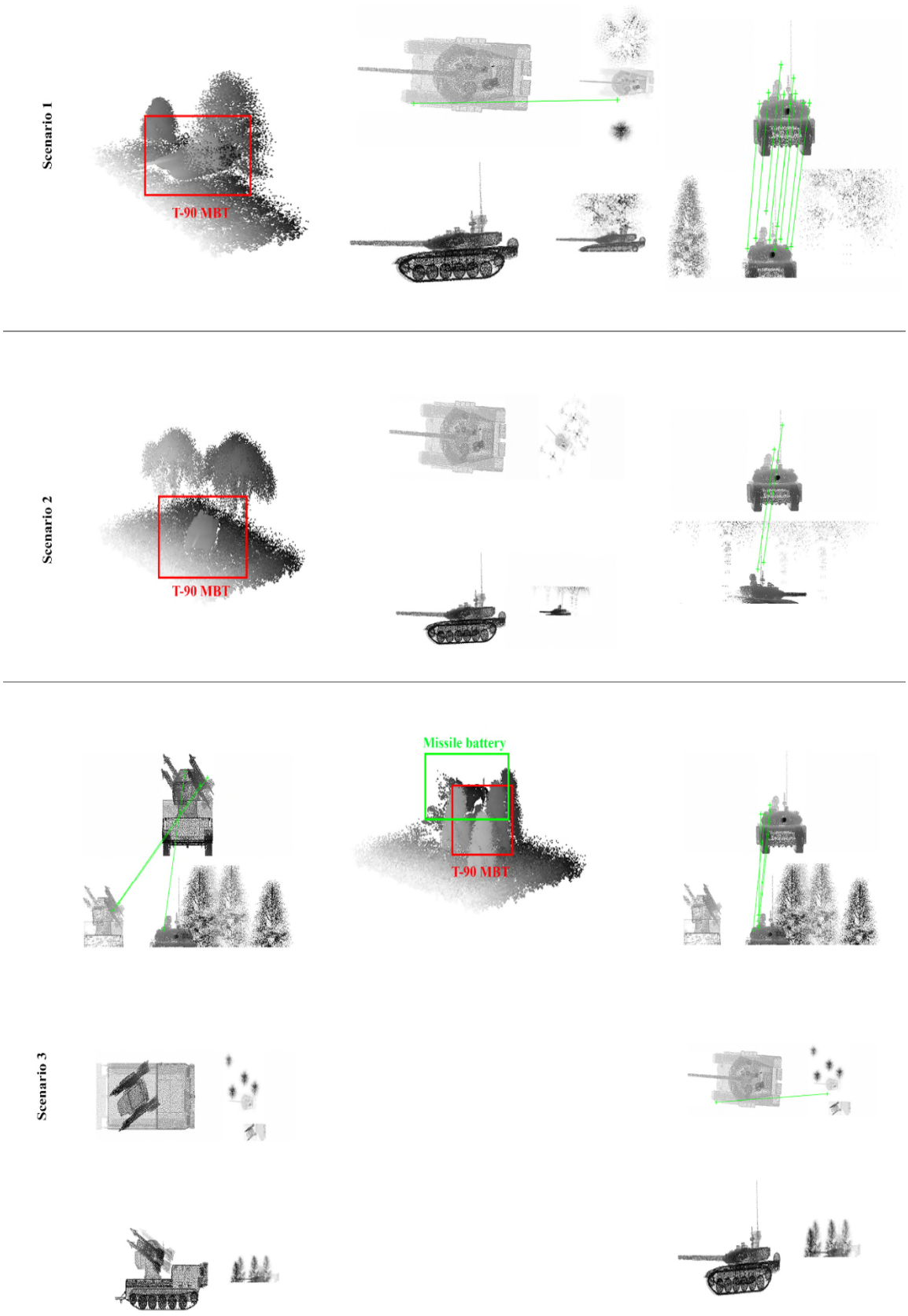


Figure 3- 20 SPR applied on various forestry scenarios (image from [20]**)**

3.2.6.4 Comparison with the Rotational Projection Statistics (RoPS) algorithm

In the following trials SPR is challenged against the state-of-the-art 3D local feature based descriptor RoPS [59], which outperforms the Spin Image, THRIFT and SHOT recognition techniques [57].

The first batch of trials uses the optimal parameters of RoPS as defined by its authors [61] i.e. 5000 keypoints are randomly selected in the model object and 1000 in the scene. For these keypoints, the RoPS features are calculated and then matched via an NNDR criterion. Matches produce a transformation hypothesis that is verified via an ICP scheme. Finally, based on the verified Transformation the model is segmented from the scene.

For a LIDAR based missile, the segmentation and pose estimation subroutines are time-consuming processes that are non-mandatory. Hence, the segmentation capability, the transformation hypothesis generation and verification processes are substituted with a matching quality criterion to speed up RoPS and make it more appropriate for military type oriented ATR. This quality measure considers as the correct template match the one that provides the smallest average Euclidean feature distance with the scene. This modification maintains the matching quality of RoPS and discards the pose estimation capability, which is not mandatory for LIDAR based missiles. Hereafter, this RoPS configuration will be named as RoPS (5000-1000).

Trials consider the same database and experiments as in Section 3.2.6.2 but restrain them to the observation scale *s* because RoPS has a limited scale invariance up to *x2* [59], [167]. RoPS (5000-1000) achieves an average recognition performance of 96.4% and the processing time per pose is 118.7s exceeding by far the time constraints of a LIDAR based missile application. The reason is the time-consuming LRF calculation and the large number of keypoints and therefore features that must be matched. Focusing on the average recognition capability, SPR is marginally higher than RoPS (5000-1000) by 0.1% but most important, it is 253 times faster. In contrast to SPR, RoPS is scale invariant up to *x2* while a greater scale invariance is a mandatory demand for

missile type ATR. Therefore, it can be concluded that SPR is more appealing than RoPS as it combines high quality recognition performance, processing efficiency and greater range of scale invariance.

To speed up RoPS the number of keypoints are optimized to achieve a balance between recognition performance and computational efficiency. The equilibrium is set at matching 10 keypoints of the scene to 2000 from each model. This provides to RoPS a speedup of *x23* while a notable recognition performance is still maintained. Hereafter this RoPS configuration will be named as RoPS (2000-10). This version of RoPS is evaluated under the same transformations and perturbations as in Section 3.2.6.2 at scale *s*. On average RoPS (2000-10) achieves 80% recognition performance in 7.2s. In contrast, the proposed SPR solution gains a recognition rate of 96.5% while in parallel it is still x15.6 faster. Figure 3- 21 presents a detailed SPR and RoPS (2000-10) comparison per target and trial. In all trial and target combinations, SPR achieves a higher recognition rate with a large margin except for the missile battery target under combined 3D rotation, noise and sparsity. Figure 3-22 compares SPR with both RoPS variants where it is evident that SPR has the same ATR performance as RoPS while being two orders of magnitude faster.

Finally, SPR has a notable lower memory demand to store the templates compared to its RoPS based competitors. SPR requires 380KB/template on average, while RoPS (5000-1000) 5400KB/template and RoPS (2000-10) 2160KB/template.

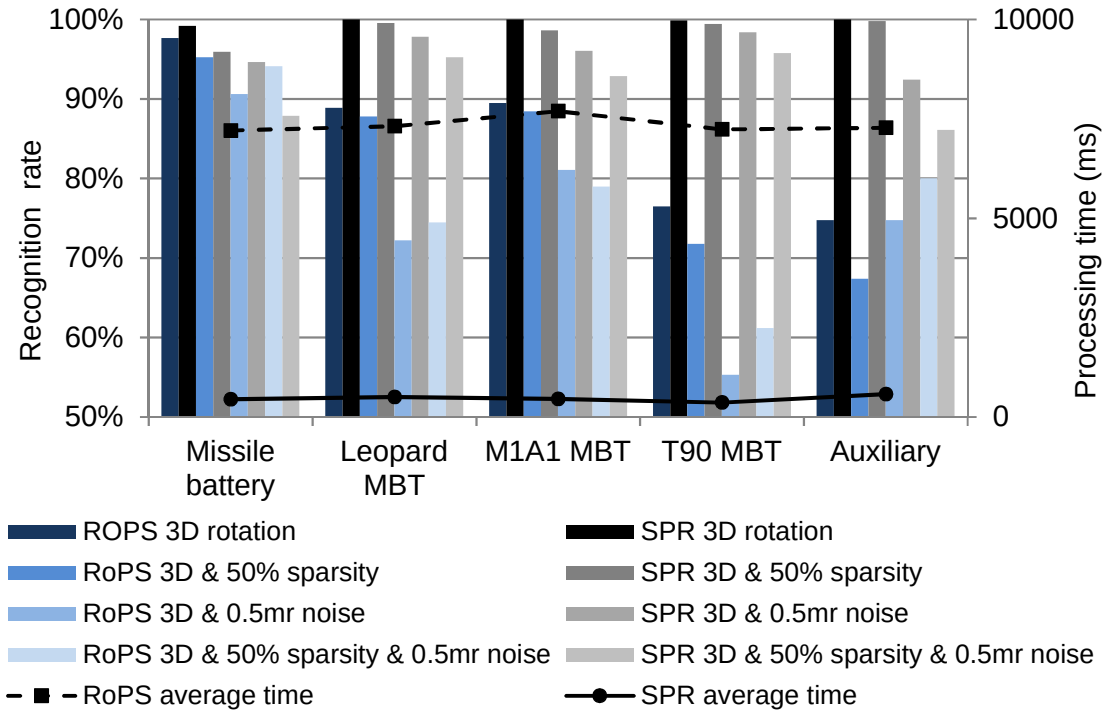


Figure 3- 21 SPR and RoPS (2000-10) comparison. Graph shows the average processing time and bar plot the recognition performance

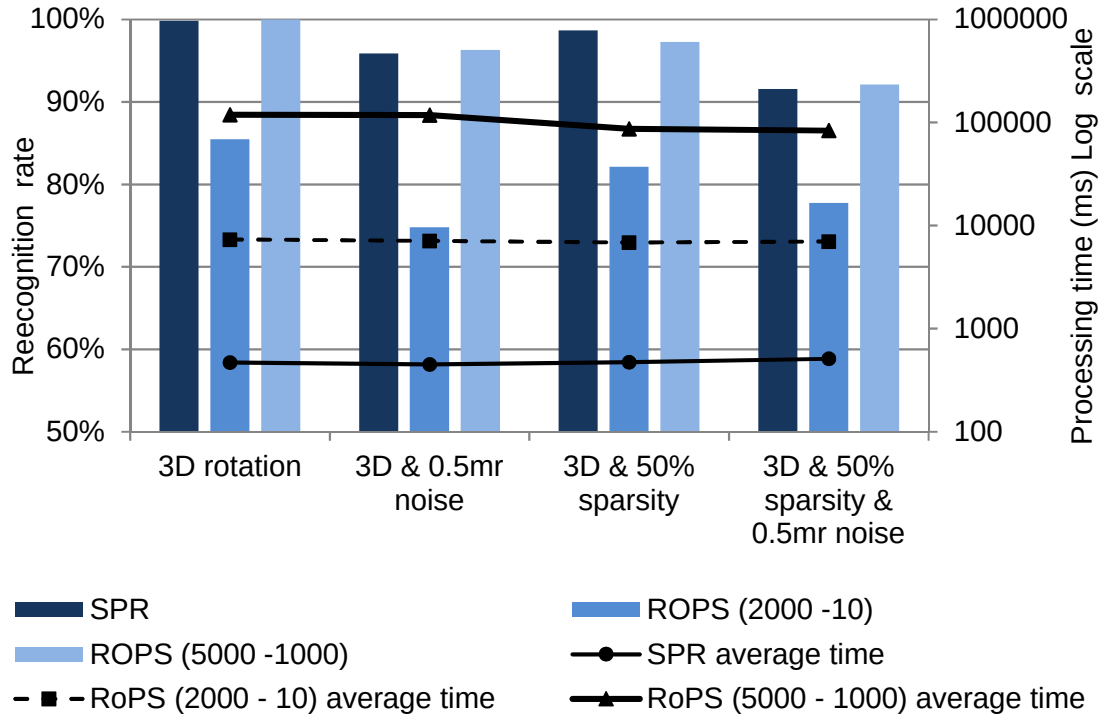


Figure 3- 22 Comparison between SPR and both RoPS variants. Graph shows the average processing time and bar plot the average recognition performance

3.3 Conclusion

In this chapter, the author proposes the SURF Projection Recognition (SPR) solution, which is a computationally efficient 3D ATR algorithm robust to rigid transformations and various perturbations. Specifically, SPR is robust to 3D rotation combined with scale change, Gaussian noise, and sparse target representation. Appealing features of SPR are the combination of high recognition performance, fast execution time and low storage memory demand.

SPR meets time restrictions by discretizing the initial point cloud and decomposing the 3D recognition problem into three 2D ones. In addition, the required database entries per target are reduced to the minimum of one pose per target, which is considered as a massive reduction compared to a multi pose and multi azimuth approach that is the norm in ATR systems. The resulting 2.5D projections are then processed using an extension of the SURF algorithm, which is named SPR. Trials on pose, scale and obscuration tolerance against various target types and in various scenarios show that the SPR technique has a high recognition rate and is highly processing efficient. Although it has a comparable ATR performance to the original implementation of RoPS, it is two orders of magnitude faster and is fully scale invariant. Finally, SPR's storage memory demand is substantially lower by a factor of x14.2 and 5.7 compared to RoPS (5000-1000) and RoPS (2000-10) in respect.

The high performance and low processing time of SPR solution can be explained by the following three facts:

a. SPR achieves 3D rotation invariance due to the complimentary nature of the three range images.

b. Robustness to scale is possible due to the resizing strategy applied to each range image.

c. SPR can successfully handle perturbations like noise and target subsampling due to combining a resizing strategy and discretizing the point cloud.

Linking the SPR's performance to current military tactics, it is concluded that:

a. Pose independence is an important factor for land based anti-armour missiles as they usually fly towards the target getting a downward but side-on or end-on view. In the late phase of engagement, they then must pop-up to perform a top attack where the armour is thinnest. Thus, the view the seeker head sees changes when the target is very close compared to that seen at longer ranges. The SPR technique is fairly pose and scale independent and hence suitable for this.

b. LIDAR has good smoke obscurant penetration and if combined with ATR using SPR would probably render it fully ineffective against LIDAR SPR type seeker heads.

c. Most anti-shipping missiles aim for the centre of mass, but approach the target at wave height, thus the target is seen from this pose. If there is a rogue wave, they will perform a pop-up to avoid it, which will suddenly change the viewpoint. Linking SPR to missile gyroscope data may alleviate this problem compared to the disturbance suffered by conventional techniques.

Despite SPR being an appealing range image based 3D ATR algorithm, its performance is affected by the depth values of the target within the scene. For the forestry scenarios presented in Section 3.2.6.3 this was compensated by forcing a ground and tree top rejection scheme and given the average known height of the targets. Despite that, accurately discarding the ground and tree tops can be quite challenging.

4 Global Based 3D ATR

THIS chapter analyses one of the two main 3D descriptor categories, the Global feature based, while the second one, the Local feature based, is analysed in Chapter 5. As already introduced in Section 2.4.2.2, examples of Global based techniques are the Shape Distributions [105], VFH [106], CVFH [28], OUR-CVFH [32], ESF [27], the Compressed VFH [107], the 3D Feature Maps [108], the Geodesic Eccentricity method [109] and GOOD [110]. The contribution of this chapter is the Projection Density Energy based solution [13]. Figure 4-1 presents various Global 3D ATR techniques a selection of which will be analysed and discussed in the following sections.

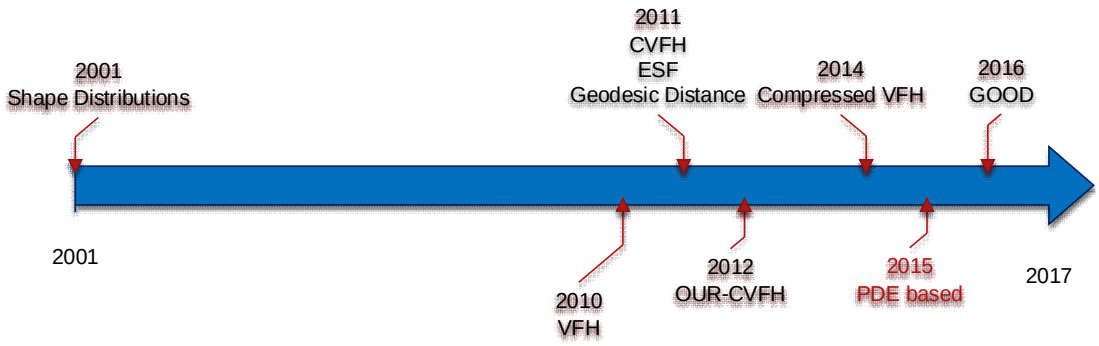


Figure 4- 1 Current Global 3D descriptors

4.1.1 Shape distributions

Osada in [105] describes the entire object with geometric shape functions that are based on simplistic yet descriptive measurements of angles, distances, areas and volumes. Osada proposes five shape distribution functions namely the:

a. A3 which relies on the angular measurement of three random points P_i,P_j,P_k $\{i,j,k \mid i,j,k \in \mathbb{N} \wedge \text{i,j,k<K}\}$, with K the total model points.

$$A_3 = \langle P_i, P_j, P_k \rangle \quad \textbf{(4- 1)}$$

b. D1 which measures the Euclidean distance of a random point P_i to the centroid of the model.

$$D_1 = \left\| \frac{1}{K} \sum_{i=1}^{K} P_i - P_i \right\|_2 \quad \textbf{(4- 2)}$$

c. D2 which measures the Euclidean distance between two random point pairs P_i,P_j

$$D_2 = \left\| P_i - P_j \right\|_2 \quad \textbf{(4- 3)}$$

d. D3 which triangulates random points P_i,P_j,P_j and measures their square root area Δ

$$D_3 = \sqrt{(\Delta_{PiPjPk})} \quad \textbf{(4- 4)}$$

e. D4 which randomly selects four points P_i,P_j,P_k,P_m $\{i,j,k,m \mid i,j,k,m \in \mathbb{N} \ \wedge \ \mathrm{i,j,k,m<K}\}$ and measures the cube root of their volume V

$$D_4 = \sqrt[3]{(V_{PiPjPkPm})} \qquad \textbf{(4- 5)}$$

Feature matching is based on a L1-norm metric between the template and the target corresponding shape distributions. According to its authors, all shape descriptors are processing efficient and robust to rigid transformations, noise and minor object subsampling. The Shape distribution that positively stands out is the D2 with an example shown in Figure 4- 2.

Although Shape distributions are quite simplistic, computational efficient and robust in various deformations and distortions e.g. noise and scale [105], [168] they have a number of disadvantages. Mainly, statistics are sampled over the entire object neglecting any kind of shape property distribution [106]. Therefore, two completely different objects might create the same shape distribution [168].

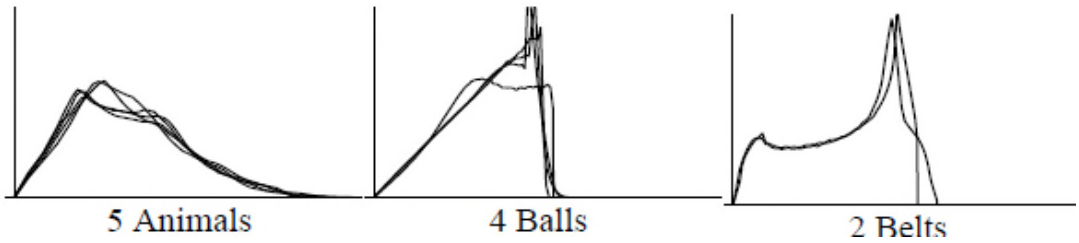


Figure 4- 2 The D2 Shape Distribution of various objects (image from [105]**)**

4.1.2 Ensemble of Shape Functions (ESF)

Given the promising performance of the Shape distributions, Wohlkinger [27] proposed a variant that encompasses histograms based on the A3, D3, D2 distributions along with a point pair distance ratio metric *R*. The three former shape distributions are further divided into sub-histograms based on whether a point pair or triplet, depending on the shape distribution, belong or not on the

model's surface. Hence the *ON*, *OFF* or *MIXED* cases can occur. The *ON* condition includes the cases where the connecting line of the point-pair to be encoded is lying on the surface of the object, the *OFF* where only the endpoints are on the surface and the connecting line is off the surface while the *MIXED* condition, where the line is partially on and off the surface. The point pair distance ratio metric *R* is based on a pairwise Euclidean distance of randomly selected vertices P_i, P_j. and concerns the ratio of the distance belonging to free space to the distance lying on the model:

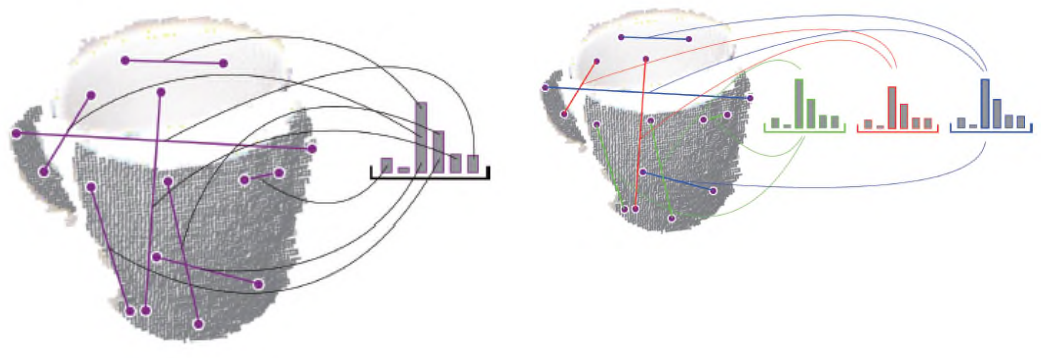
$$R = \frac{\left\| P_i - P_j \right\|_{2\ free\ \text{space}}}{\left\| P_i - P_j \right\|_{2\ on\ \text{model}}} \qquad \textbf{(4- 6)}$$

Finally, the three shape distributions including the three sub-cases (*ON*, *OFF*, *MIXED*) and the distance ratio metric *R* are converted into histograms that are concatenated to form the 640-element long ESF descriptor:

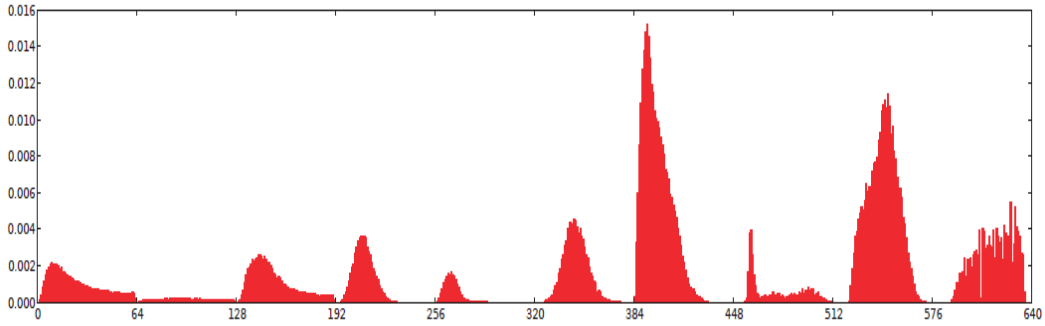
$$ESF = A_{3c} \,||\, D_{3c} \,||\, D_{2c} \,||\, R \qquad \textbf{(4- 7)}$$

where $c \in \{ON, OFF, MIXED\}$. Figure 4- 3 depicts the ESF descriptor.

Although ESF has a good object recognition capability, it requires 80 template views per target such as to cover a wide range of possible target poses. This extended template size increases the total storage memory requirement and matching time, which are both limited on-board missile platforms.



(a) (b)



(c)

Figure 4- 3 ESF descriptor (a) D2 Shape Distribution (b) D2 Shape Distribution with *ON* (Green), *OFF* (Red) and *MIXED* (blue) sub-histograms (c) ESF descriptor (image from [27]**)**

4.1.3 Viewpoint Feature Histogram (VFH) Group

4.1.3.1 VFH

VHF [106] is the local Fast Point Feature Histogram (FPFH) 3D descriptor variant extended in the global domain. The VFH descriptor has a Viewpoint and an extended FPFH component. The latter comprises of a global FPFH descriptor (by extending the description radius to include the entire object) that encodes the pairwise angles:

$$\alpha = v \cdot n_j \qquad \beta = n_i \frac{P_c}{\|P_c\|} \qquad \phi = u \cdot \frac{P_i - P_c}{\|P_i - P_c\|} \qquad \theta = \arctan(w \cdot n_c, u \cdot n_c) \qquad \textbf{(4- 8)}$$

where *u, v, w* represent the axes of a local reference frame at P_i set by:

$$u = n_c \qquad v = u \times \frac{P_i - P_c}{\|P_i - P_c\|} \qquad w = u \times v \qquad \textbf{(4- 9)}$$

while n_i and n_c is the estimated surface normal at P_i and at the global centroid P_c respectively.

The former, the Viewpoint component, measures the angular variation between each keypoint normal and the central viewpoint direction translated to each keypoint. The VFH descriptor has a length of 263 elements i.e. 128 for the viewpoint and 45 per pan, tilt and yaw angles. The VFH descriptor is shown in Figure 4-4.

Major drawbacks of VFH are:

a. Noise and target occlusion affect the establishment of the reference frame which in turn influences the entire VFH descriptor and thus its recognition performance.

b. It requires 450 templates per target that substantially increase matching time and descriptor's storage memory required for database storage.

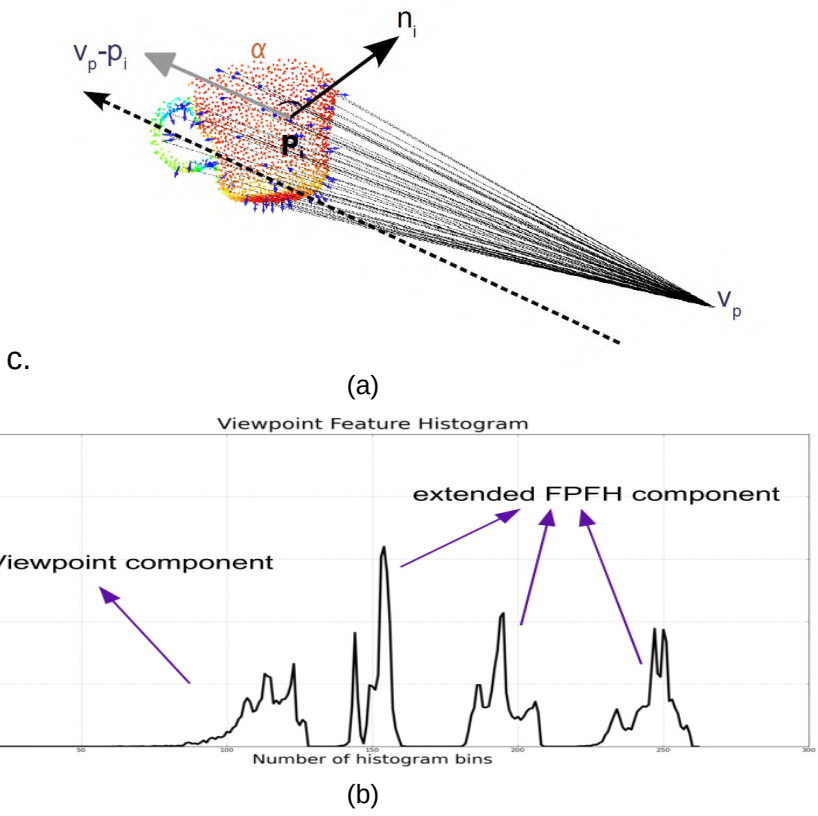


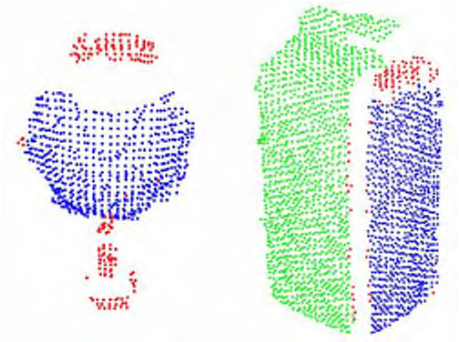
Figure 4- 4 (a) Viewpoint angular variation (b) VFH descriptor (image from [106]**)**

4.1.3.2 Compressed VFH

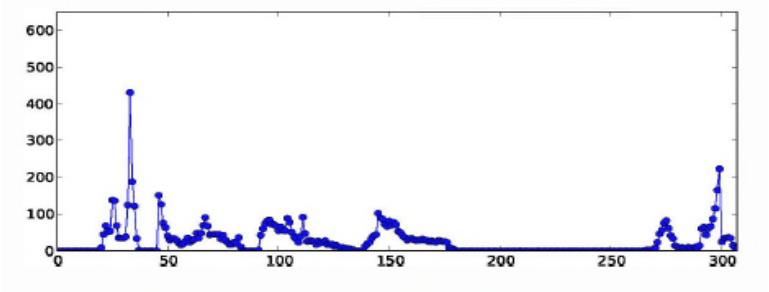
Salih *et al.* [107] propose a compressed variant of the VFH descriptor to reduce the length of the descriptor and to compensate for missing parts of the scene object. Since the VFH descriptor is quite sparse, they apply eigenvalue decomposition to extract the dominant features instead of exploiting the entire sparse histogram.

4.1.3.3 Clustered VFH (CVFH)

In contrast to VFH, the CVFH [28] descriptor establishes the reference frame based only on clustered stable parts i.e. smooth and continuous regions of the object and not on the entire object as VFH does. Hence P_c of equations 4-8 and 4-9 are substituted with $\boldsymbol{P_{cc}}=\{P_{c1},P_{c2},\ldots\}$ depending on the clustered region involved in the calculations. This methodology aims at enhancing robustness to noise and occlusion. Even though the reference frame is individually set on each clustered part of the object, the CVFH descriptor is assembled on the entire object. Figure 4-5 shows an example of the CVFH descriptor. Despite CVFH having better performance compared to its predecessor the VFH, misalignment errors of the reference frame still exist affecting recognition performance.



(a)



(b)

Figure 4- 5 (a) surface clustering (b) CVFH descriptor (image from [28]**)**

4.1.3.4 Oriented Unique and Repeatable CVFH (OUR-CVFH)

A further evolvement of CVFH is proposed by Aldoma [32] who establishes one reference frame per stable cluster of points based on a different reference frame estimation method.

Specifically, for a given keypoint $Pc,\{c \mid c \in \mathbb{N}, c \le K\}$ and a radius r_c, a spherical volume $\boldsymbol{P_i}$ is extracted containing the verctices P_i. For each $\boldsymbol{P_i}$ the eigenvalues $Cv_j=\lambda_j$, $j \in \{0,1,2\}$ are calculated, where λ_j is the j^{th} eigenvalue of the weighted covariance matrix C, and v_j is the j^{th} eigenvector. The weighted covariance matrix is given by [52]:

$$C = \frac{1}{\sum_{i:d_i \le R}(R - D_i)} \sum_{i=1}^{k} (R - D_i)(P_i - P_c)(P_i - P_c)^T \qquad \textbf{(4- 10)}$$

with $D_i = \left\| P_i - P_c \right\|_2$ and R the distance of P_c to the furthest P_i. Sign disambiguation for rotation invariance is achieved through selecting the sign of an eigenvector such as to render it coherent with the majority of the vectors it represents. This procedure is applied to the eigenvector associated with the smallest eigenvalue defining the z axis while the x and y axes are at right angles.

The advantage of OUR-CVFH is being robust to missing vertices because it establishes one descriptor per smooth region. Nevertheless it still requires 12 views per object to facilitate a good recognition performance [32]. Figure 4-6 depicts the OUR-CVFH concept.

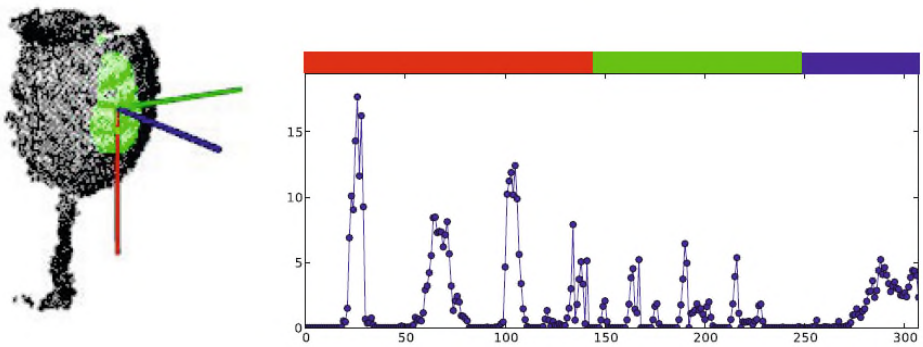


Figure 4- 6 OUR-CVFH. The coloured regions of the descriptor indicate the corresponding axis used for alignment (images from [32]**)**

4.1.4 Discussing current Global Based 3D descriptors

Even though current 3D global based descriptors are processing efficient operating in the order of milliseconds [27], [169], they have a few drawbacks affecting military type applications. Specifically:

a. Current solutions mainly originate from two techniques, namely the Shape distribution and the 3D local based descriptor FPFH which has an extended description radius to include the entire object. Despite the evolution of current global based descriptors, they still suffer from the constraints of their originating descriptor. Although FPFH will be analysed in Section 6.1.6.1., for completeness it is mentioned at this point that FPFH suffers from low robustness to clutter, occlusion and noise [90], [112]. The interrelationship among the descriptors is presented in Table 4- 1.

b. FPFH based solutions i.e. VFH, CVFH, OUR-CVFH and compressed VFH require a reference frame which increases the total computational burden.

c. Robustness to occlusion and rotation is achieved by using 12 up to 450 template images (views) per target to be recognised. The exact number depends on the descriptor. This is a major deficiency of current global descriptors as matching time and storage memory requirements increase substantially. Details on the template size requirements are shown in Table 4-1.

d. Current global based approaches are designed to recognise objects that comprise of distinguishable primitive shapes i.e. concave, convex or planar objects such as household objects. The characteristics of these objects is much simpler compared to highly complex surfaces found in battlefield scenarios.

e. Up-to-date descriptors are designed to meet the needs of robotic applications. Although the orientation estimation is useful for such applications, for the missile ATR case this is appealing but not mandatory. Indeed, the guidance unit of the missile needs only to know if the missile platform is aiming at the correct target or not. Therefore, estimating the

orientation of the target consumes valuable processing resources and omitting it can be useful.

f. The high recognition performance reported by global descriptors is obtained from accumulating the recognition rates of *n*-nearest neighbour matches (n-NN) instead of 1-NN which is mandatory for a missile based application such as to reduce false targeting.

g. Global based descriptors have only inter-class recognition capabilities and current algorithms are evaluated only on simplistic objects.

Table 4- 1 Global based 3D descriptors

Name	Length	Template views	Predecessor descriptor
Shape distribution	64	not reported	---
ESF	640	80	Shape distribution
VFH	263	450	FPFH
Compressed VFH	Variable	450	FPFH, VFH
CVFH	308	12	FPFH, VFH
OUR-CVFH	308	12	FPFH, VFH, CVFH

Driven by those drawbacks this research proposes a computationally efficient global 3D ATR algorithm suitable for military type LIDAR based time-critical applications with limited hardware capabilities. This contribution relies on information theory concepts that are combined with a Constant False Alarm rate (CFAR) adaptive threshold and applied on multiple 2D projections of the 3D object. The proposed descriptor is invariant to rigid transformations combined with Gaussian noise and uniform target subsampling. It should be noted that

although the suggested methodology focuses on missile based 3D ATR, it can be exploited on commercial datasets as well. In contrast to the simplistic datasets on which current algorithms are evaluated on, the proposed solution is applied on real targets from the UWA dataset and on military targets from the Princeton shape benchmark. The object class recognition performance achieved is more than 90% in less than 100ms for point clouds exceeding 90,000 points. In addition, this technique is challenged with the state-of-the-art 3D local based descriptor RoPS and trials reveal that the proposed algorithm achieves a higher recognition rate, two orders of magnitude faster execution time and one order of magnitude lower descriptor storage memory demand.

4.2 Fast 3D Object Matching with Projection Density Energy

This method transforms the 3D problem into multiple 2D ones based on 2.5D projections. Each projection undergoes a statistical analysis relying on the Projection Density Energy (PDE), while large pose variations between the target and the template are compensated with a CFAR based threshold. Finally, template matching relies on a cost function, leveraging information from each 2.5D projection. Although the proposed descriptor is computationally efficient, processing time is further reduced by exploiting a single 3D template per target.

4.2.1 Projection Density Energy based algorithm

Given a point cloud $\boldsymbol{P} \in \Re^3$, each vertex can be represented as $P_u, \{u \mid u \in \mathbb{N}, u \le M\}$ where M is the total number of points. Initially P_u is uniformly quantized to P_{qu} with a quantization step Δ to reduce the amount of points and thus the overall processing time:

$$P_{qu} = sign(P_u)\,\Delta\left\lfloor \frac{|P_u|}{\Delta} + \frac{1}{2} \right\rfloor \qquad \textbf{(4- 11)}$$

Similarly to the suggested SPR algorithm presented in Section 3.2, the value of Δ is experimentally chosen such as to balance the algorithm's recognition performance and processing efficiency.

Computational complexity is further reduced by transforming the 3D recognition problem into multiple sub-dimensional ones [170], [171] incorporating though information from the 3D world. Therefore, each quantised point $P_{qu},\{qu \mid qu \in \mathbb{N}, qu \le L, L < M\}$ is orthographically projected on each plane of a *XYZ* GRF. The latter is set on the LIDAR sensor with axes fixed parallel to its physical width, height and depth dimensions. Projections are based on the orthographic projection matrix P_{ortho} by zeroing the appropriate binary remapping coefficients $c_1, c_2, c_3 \in \{0,1\}$ depending on the plane the cloud will be projected. For example, if $c_1 = c_2 = 1$ and $c_3 = 0$ then the X-Y projection is received. In parallel, the point cloud is translated to the origin of the GRF by applying the proper translation coefficients t_1, t_2, t_3. The coordinates $\tilde{P}$ of the orthographically projected point cloud after being quantized and translated to the origin of the GRF are given by:

$$\tilde{P} = \begin{bmatrix} \tilde{x}_{qu} \\ \tilde{y}_{qu} \\ \tilde{z}_{qu} \\ 1 \end{bmatrix} = P_{ortho} P_{qu} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{qu} \\ y_{qu} \\ z_{qu} \\ 1 \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ 1 \end{bmatrix} \quad \textbf{(4- 12)}$$

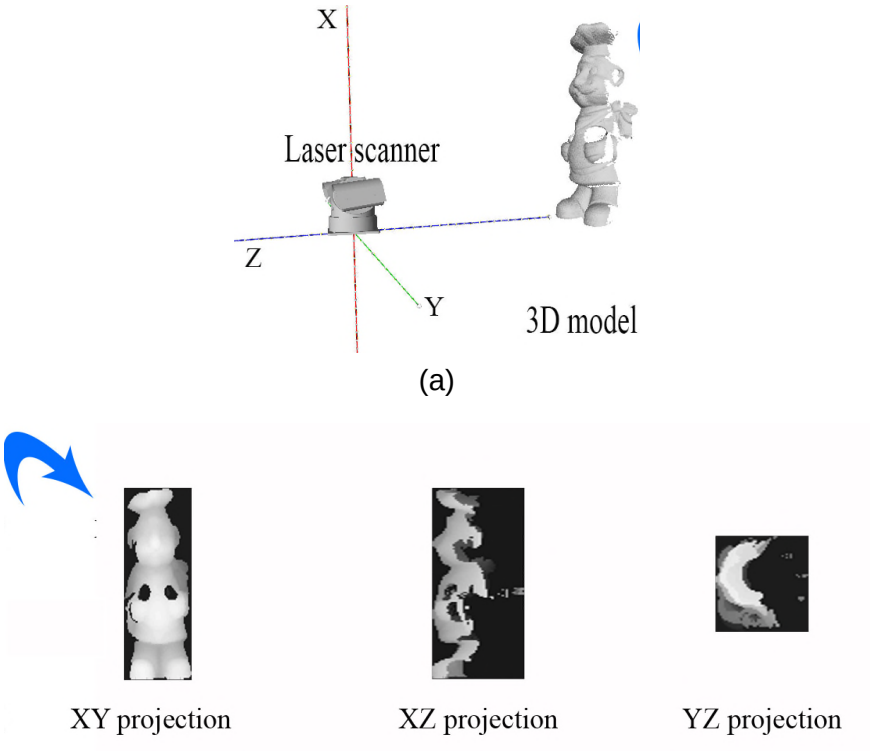
where $\tilde{x}_{qu}, \tilde{y}_{qu}, \tilde{z}_{qu}$ are the coordinates of the orthographically projected points on the YZ, XZ and XY plane respectively. The three orthographic projections are 2.5D range images i.e. simplified versions of the 3D point cloud P_{qu}. In these images, the depth value of each plane e.g. $I(\tilde{x}_{qu}, \tilde{y}_{qu}) = \tilde{z}_{qu}$ is unique and represents the distance between the target and the LIDAR seeker. Figure 4-7 presents an illustration of the reference frame conversion and the 2D projections.

4.2.2 Projection Density Energy

The next stage of the suggested global descriptor involves calculating the Projection Density Energy (PDE) [116] of each of the three $I(\tilde{x},\tilde{y})$ 2.5D image projections. PDE is based on Shannon Entropy and measures the distribution of the non-zero values of each range image *I* of size *mxn* normalised by the number

of non-zero elements N of the corresponding $I(\tilde{x},\tilde{y})$ projection. For further processing efficiency, the Taylor-series expansion is exploited [116]:

$$PDE \approx -\sum_{\substack{1\le x\le m\\ 1\le y\le n}} \left(\frac{I(\tilde{x},\tilde{y})}{N} \left(\frac{I(\tilde{x},\tilde{y})}{N} - 1 \right) \right) \qquad \textbf{(4- 13)}$$



(a)

(b)

Figure 4- 7 Range image of (a) real model from the UWA database (b) quantized and orthographically projected onto the planes of the LIDAR based GRF (image from [13])

Although Shannon Entropy is not a correlating criterion between objects as cross-correlation is, it nevertheless provides a quick indication of the object's dissimilarity. $\Phi(I)$ is defined as:

$$\Phi(I) = \frac{1}{3}\left(\sum \left| PDE_i^M - PDE_i^S \right| \right) \qquad \textbf{(4- 14)}$$

where PDE_i^M, PDE_i^S are the *PDE* for the model and the scene image per projection plane *i={XY,XZ,YZ}*. Based on the $\Phi(I)$ metric, the smaller the $\Phi(I)$ more similar the target-template are.

4.2.3 Cost function

In contrast to VFH, CVFH, OUR-CVFH and ESF that require 80, 450, 12 and 12 views per target respectively (Table 4-1) the PDE algorithm uses a single template per target model. This strategy provides enhanced computational efficiency but prohibits $\Phi(I)$ from being out-of-plane rotational invariant. Therefore, 3D rotational invariance is compensated by introducing a cost function:

$$\Im(I) = \Omega(I) \cdot \Phi(I) \cdot S(I) \qquad \textbf{(4- 15)}$$

where *S(I)* is a scale factor and $\Omega(I)$ the average pairwise target - template binary projection difference.

4.2.4 Scale factor estimation

As the target rotates in 3D, some of its parts shift from the background to the foreground and vice versa. This effect is also evident on each of the 2.5D projected images forcing a local zooming effect as presented in Figure 4- 8.

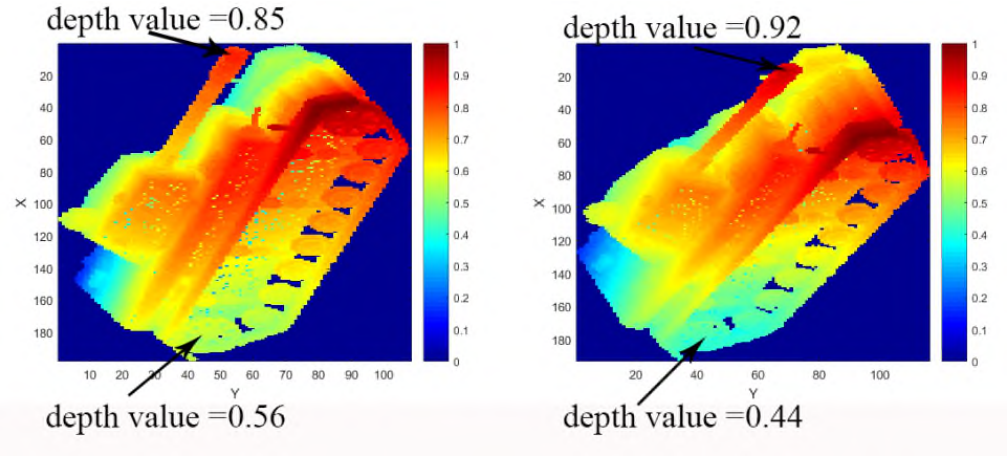


Figure 4- 8 Local zooming effect on the 2D plane projection due to out-of-plane rotation of the target (colour-coded for better visualization)

According to Mikolajczyk and Schmid [172] the ratio between the scales of the corresponding points where the extrema in scale space are found is the scale factor between the point neighbourhoods. The same paper concluded that most accurate results are provided by the Laplacian-of-Gaussian kernel whilst the second best, but faster to compute, are given by the Difference-of-Gaussian (DoG). Influenced by that, the characteristic scale S_{2D} of the entire 2.5D image is set as the average value of the characteristic scale of the matched keypoints μ between each 2.5D image and the corresponding template projection:

$$S_{2D} \sim \frac{1}{\mu}\sum_{1}^{\mu}\left(\arg_{local\max}\left(I(\tilde{x},\tilde{y}) * (G(\sigma_{n-1}) - G(\sigma_n))\right)\right) \quad \textbf{(4- 16)}$$

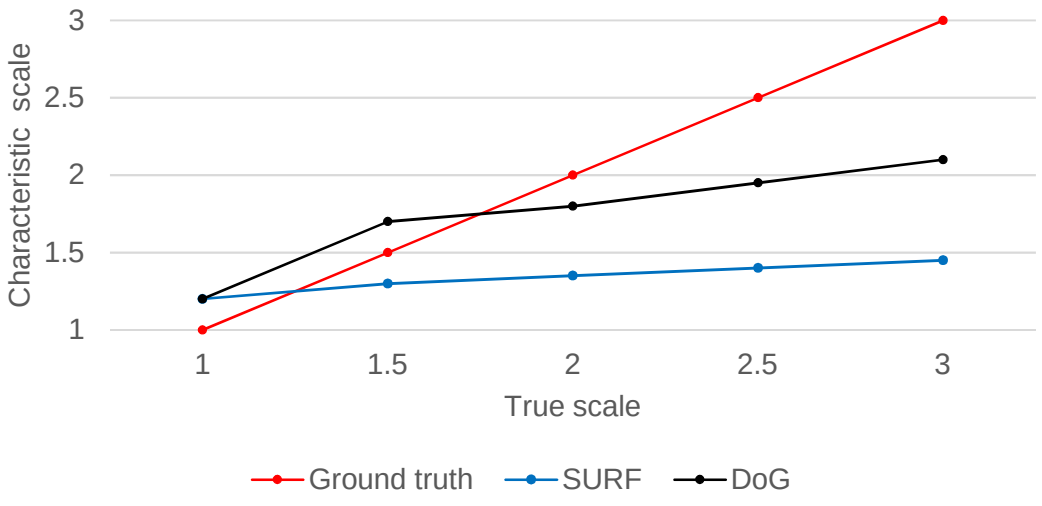
where $G(\sigma_n)$ is the Gaussian kernel with variable size and standard deviation equal to σ.

To speed-up the $S(I)$ estimation S_{2D} is relaxed to determine whether the zooming effect is increasing or decreasing, rather than estimating the true scale between the matched keypoints. Therefore instead of exploiting Equation 4-16, the DoG is approximated with the determinant of the Hessian matrix as in SURF [173]. Hence:

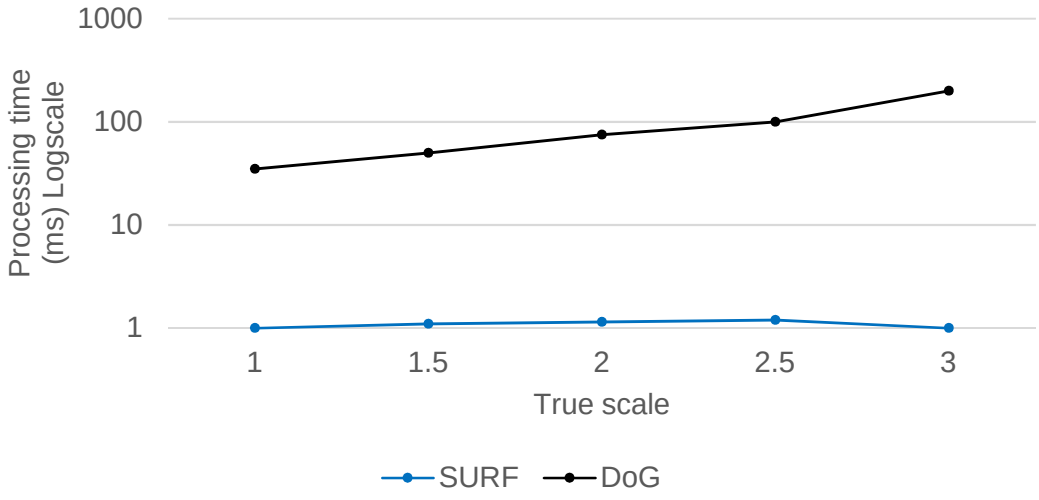
$$S_{2D} \sim \frac{1}{\mu}\sum_{1}^{\mu}\left(\arg_{local\max}\left(\det\left(H_{approx}(I)\right)\right)\right) \quad \textbf{(4- 17)}$$

where $\det(\mathrm{H_{approx}}) = \mathrm{D_{xx}D_{yy}} - (0.9\,\mathrm{D_{xy}})^2$ and $\mathrm{D_{xx}}$, $\mathrm{D_{xy}}$, $\mathrm{D_{yy}}$ are the box filters approximating the second order Gaussian derivatives as in SURF. For the 3D case and for further computational efficiency, Equation 4-17 is applied on the concatenation of the three 2.5D projections rather than in each projection individually. Hence, the characteristic scale $\mathrm{S_{3D}}$ of the 3D object is defined as:

$$S_{3D} \sim \frac{1}{\mu}\sum_{1}^{\mu}\left(\arg_{local\max}\left(\det\left(H_{approx}(XY \,||\, XZ \,||\, YZ)\right)\right)\right) \quad \textbf{(4- 18)}$$



(a)



(b)

Figure 4- 9 Comparison of SURF and DoG based approach (a) Characteristic scale estimation (b) processing time

For completeness, the suggested SURF based characteristic scale estimation is compared against the classic DoG estimation. The comparison shown in Figure 4- 9 focuses only on the trend of the scale change as this is of interest in the *S(I)* estimation. Although the DoG based approach is more accurate, the suggested technique is x70 faster and still provides the correct scale trend. In specific, the average time of the DoG is 84ms and the suggested SURF based requires only 1.2ms. Detailed results are.

Finally, the scale factor considering only the matched keypoints is defined as:

$$S(I) = \frac{S_{3D_{scene}}}{S_{3D_{model}}} \quad \textbf{(4- 19)}$$

4.2.5 Constant False Alarm Rate (CFAR) Estimation

Scenarios of interest consider recognising objects with a wide pose variation in both in and out-of-plane rotations. Using a fixed threshold to match the SURF features for the *S(I)* estimation is not an optimum solution because the larger the target's out-of-plane rotation, the fewer the correspondences. Thus, in contrast to the majority of the 2D/3D pattern recognition literature that suggest a fixed keypoint matching threshold for the NNDR matching criteria [2], [26], [32], [49], [51], [53], [58], [61], [63]–[65], [67], [68], [86], [92], [96], [112], [135], [146], [151], [173]–[185], an adaptive one is used that can provide keypoint matches even under large out-of-plane rotations. In specific, the CFAR [186] radar concept is used that establishes a variable threshold aiming at a fixed false alarm rate. It should be noted that a variable matching threshold affects the quality of the matched keypoints but in any case provides the best possible matches. Despite that, SURF matches do not aim at high quality keypoint matching but only in the $S(I)$ estimation. On the contrary a fixed low NNDR threshold value would provide mixed good and bad quality keypoint matches.

Calonder *et al*. [94] present distributions of Hamming distances for matching point-pairs described by the binary descriptor BRIEF. Loosely extending that from the binary into the floating-point domain by considering that each element of the SURF descriptor is a single bit, the pairwise Hamming distance between all target and template SURF features are calculated to identify the fraction of bits that they disagree. Adding to that, Daugman [187] declares that comparisons between bits from different descriptors are Bernoulli trials and the latter generate Binomial distributions which in this research are substituted with a normal distribution. For this CFAR implementation the CFAR type adaptive threshold is the intersection point of the bitwise Hamming distance distribution with a fixed Gaussian distribution.

Formally, the suggested CFAR threshold requires applying SURF on each of the 2.5D projections of the scene $I^S(\tilde{x},\tilde{y})$ and the template $I^T(\tilde{x},\tilde{y})$ to establish a set of scene keypoints $\boldsymbol{P^S}$ and features $\boldsymbol{f^S}$ along with the corresponding template model keypoints $\boldsymbol{P^M}$ and features $\boldsymbol{f^M}$. Inspired from Calonder's strategy, the minimum pairwise Hamming distance of all features is:

$$d_{\text{Hamming}} = \sum \left(\min \left(f_i^M \oplus f_i^S \right) \right) \qquad \textbf{(4- 20)}$$

where $i \in \mathbb{N}$ are the indices of the matched keypoints. $d_{hamming}$ is then used to plot a Gaussian distribution

$$G_{hamming}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad \textbf{(4- 21)}$$

with $x \in [-1,1]$,

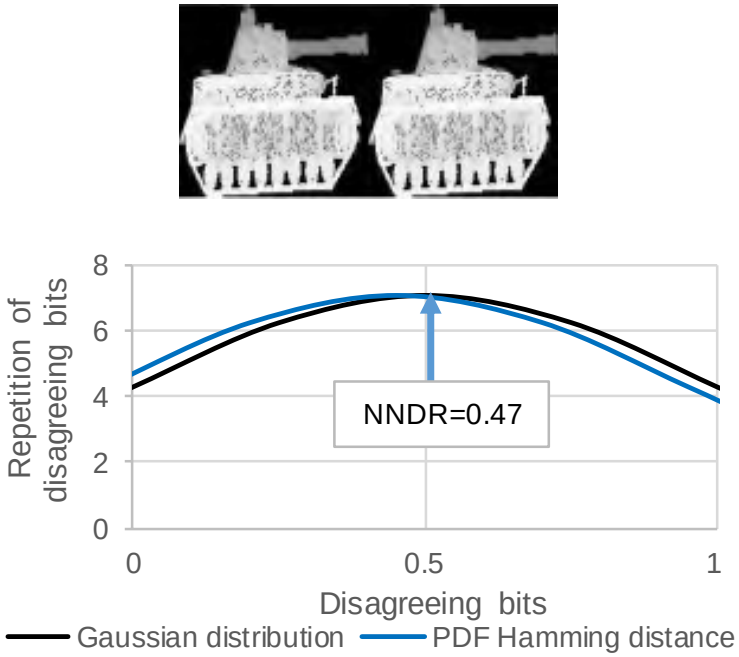
$$\mu = \frac{1}{V} \sum \left(d_{Hamming} \right), \qquad \sigma = \sqrt{\frac{\sum \left(x - \mu \right)^2}{V}} \qquad \textbf{(4- 22)}$$

where V is the cardinality of the Hamming distance vector.

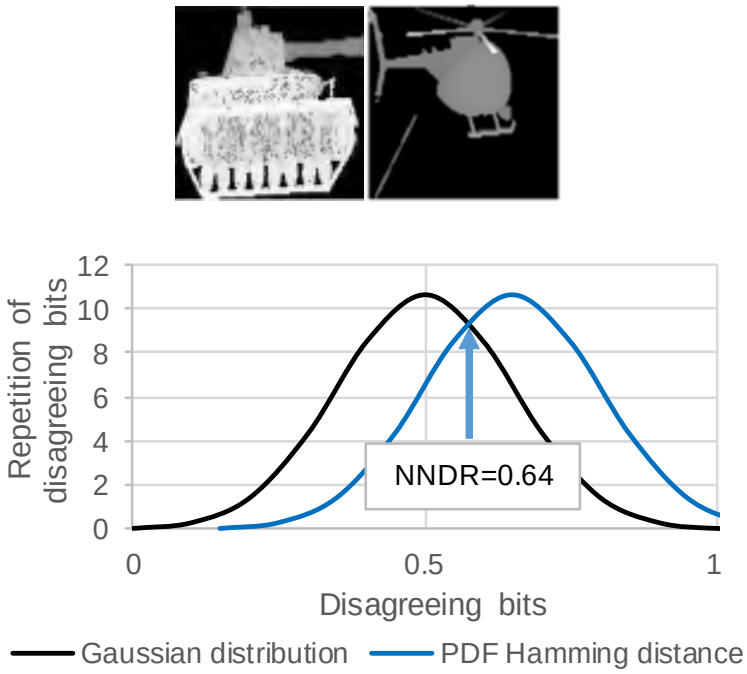
On the same graph the Gaussian distribution, G with $\mu=0.5$ and σ, is plotted to approximate the Bernoulli trials as Daugman declared. Finally, the CFAR type adaptive threshold t_{CFAR} is the smallest non-zero intersection point of the bitwise Hamming distance distribution $G_{hamming}$ with the fixed Gaussian distribution G:

$$t_{CFAR} = \arg\min(x) \quad s.t.\ G_{hamming}(x) = G(x) \ \wedge \ x > 0 \qquad \textbf{(4- 23)}$$

Figure 4- 10 presents the case where a template model is compared with a similar and with a different target. As expected the similar class requires a smaller threshold compared to the different one. In contrast to a typical fixed threshold of 0.8 [92] or 0.9 [174], the suggested CFAR approach provides a wide range of threshold values from 0.4 to 0.9 depending on the similarity of the target and the template.



(a)



(b)

Figure 4- 10 CFAR example, template – target objects along with the CFAR based NNDR threshold (a) similar objects case (b) different object case

4.2.6 Proposed recognition pipeline

This algorithm has the processing pipeline presented in Figure 4-11 and consists of an offline and an online phase. During the offline phase, the single template per model is quantised, orthographically projected onto a GRF and then SURF keypoints are detected, extracted and stored. In addition, the database includes the binary area and the PDE of each 2.5D image projection. During the online stage, the target scene undergoes the same processing done during the offline phase. Then a CFAR based threshold t_{CFAR} for the template and the target is calculated which is used during the NNDR matching stage of the SURF keypoints. The scales in which the matched keypoints are detected in combination with Equation 5-18 and 5-19 provide the scale factor *S(I)*. The latter along with the metrics *Φ(I)* and *Ω(I)* comprise the cost function $\Im(I)$. The template that provides the lowest $\Im(I)$ is considered as the match.

4.2.7 Experiments

Trials include:

a. 3D target rotation, which during trials has the notation *3D.*

b. Simultaneous 3D target rotation with Gaussian noise referred as *3D & 2mr noise*. Gaussian noise has zero mean and $\sigma = 2\overline{mr}$ where $\overline{mr}$ is the average template point cloud resolution.

c. Simultaneous 3D target rotation with a 50% uniform target subsampling noted as *3D & 50% sparse.*

d. Simultaneous 3D target rotation with $2\overline{mr}$ noise and 50% uniform target subsampling noted as *3D & 2mr noise & 50% sparse.*

e. Trials (a) – (d) but under x0.5 scale.

Even though current Global based descriptors are evaluated on simplistic objects of various datasets [188]–[191] the following trials involve highly complex structures from objects obtained from the UWA [81] and Princeton Shape Benchmark datasets [192].

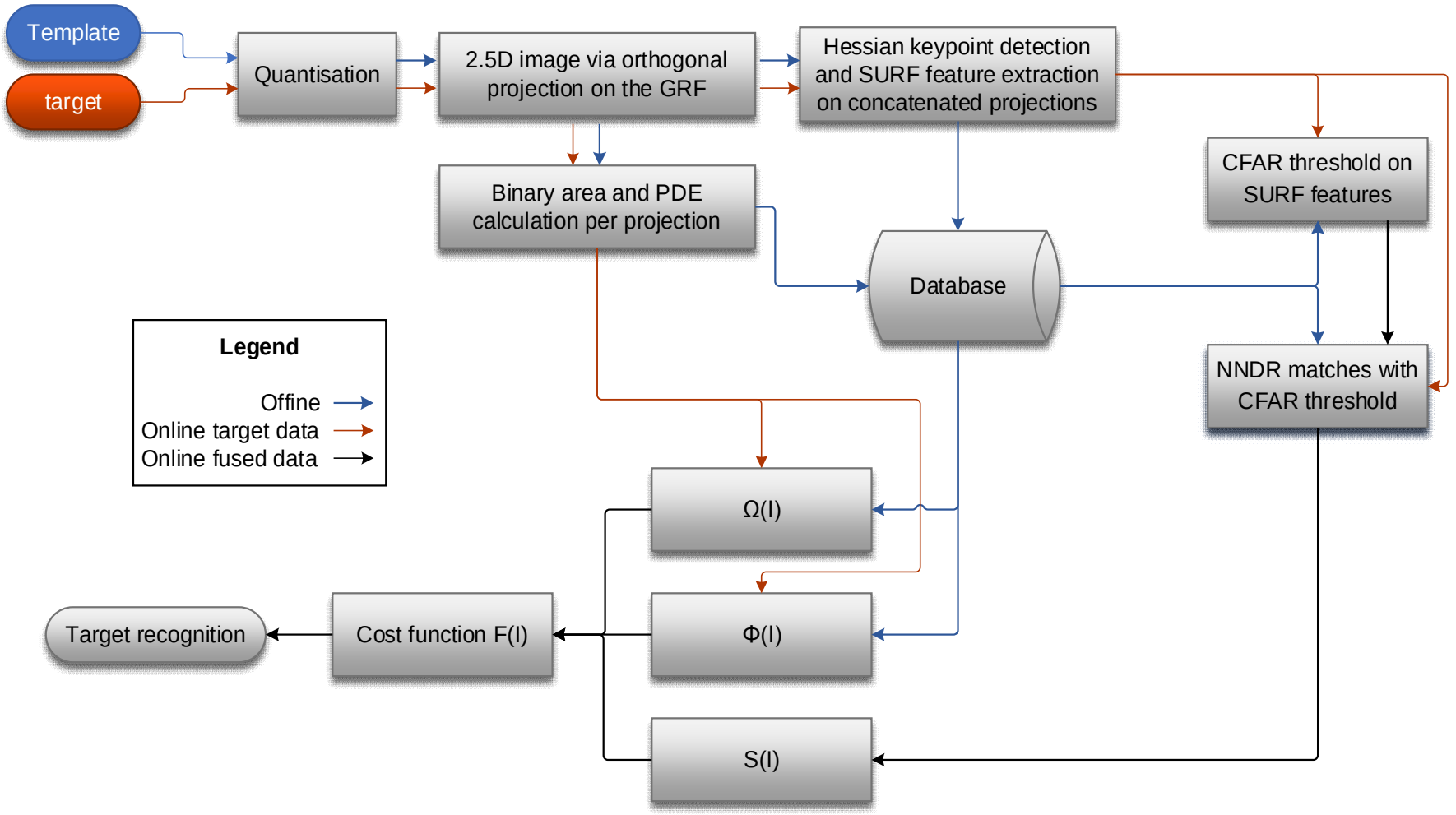


Figure 4- 11 Flow diagram of the proposed approach (image from [13])

4.2.7.1 UWA database

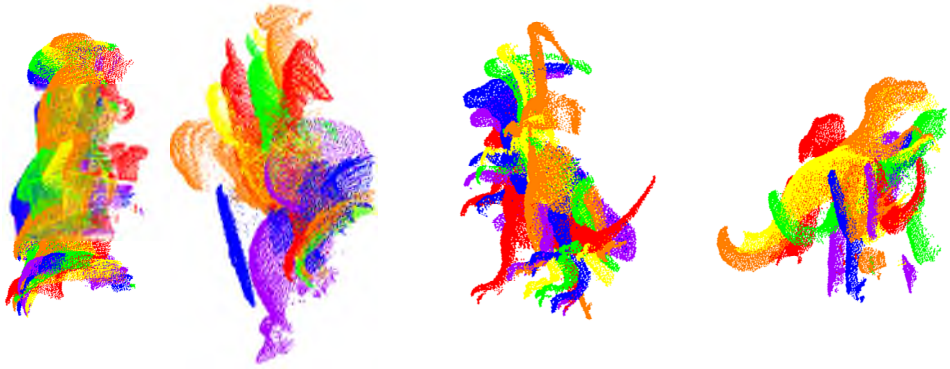
For performance comparison reasons with the existing computer vision 3D pattern recognition approaches and to increase the evaluation complexity, the first set of trials is on the UWA dataset. Although this is a non-military dataset, it is worth challenging the proposed methodology on that database because it consists of real non-ideal point clouds. The UWA dataset consists of four targets with 22, 21, 16 and 16 poses respectively while the LIDAR sensor is rotated in 3D around each target (Figure 4- 12). Scale robustness is investigated for the *x1* and *x0.5* cases.

Since the proposed approach is global based, it is limited by an inter-class recognition capability, i.e. recognition between different classes and therefore objects from the UWA dataset are grouped into three classes. These are based on a human based perception and are the *Human*, the *Bird* and the *Dinosaur*. For each object, a single viewing is used as a template (Figure 4-12).

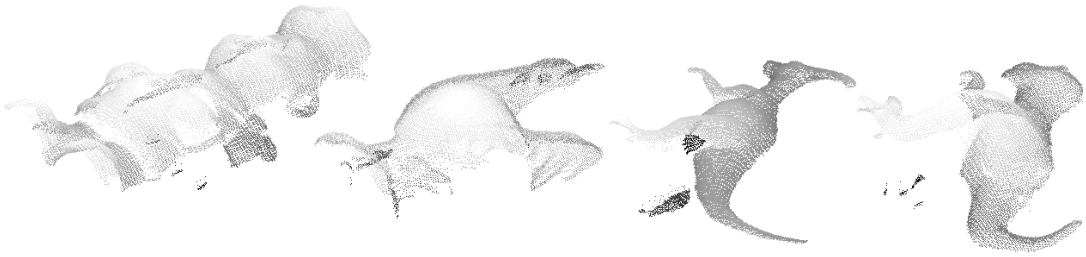
The first set of trials includes the cases presented in Section 4.2.7 (a) – (d) and the average recognition rate achieved is 91.1% in 74ms. The *Human* class has the most stable performance while the *Bird* class the least stable one. This can be explained as the former has distinct partial views consisting of many distinctive concave and convex regions, allowing a constant recognition performance of more than 90%. On the other hand, the *Bird* class comprises of a few smooth surfaces with weak depth variations and therefore CFAR provided a low threshold and so SURF keypoint matches are of low quality negatively affecting the scale factor estimation *S(I)*. Recognition performance marginally exceeds 80%, but when the computational efficiency of 91ms is taken into account, then this performance is considered as acceptable.

Despite that, when adding any type of nuisance as noise or subsampling, the smooth surfaces of the *Bird* class get corrupted. Hence, more keypoints are detected influencing the *S(I)* term and in turn the $\Im(I)$ cost function, and thus improving the recognition performance. Detailed results per target are presented Figure 4- 13.

(a)



(b)



(c)

target	Chef	Chicken	Parasaurolophus	T-rex
class	Human	Bird	Dinosaur	

Figure 4- 12 UWA dataset (a) Ideal model (not used in the trials) (b) Example of real self-occluded views Nº 1-6 (c) Database template

The next set of trials involves the cases previously challenged, but combined with scale change. Although the scale is reduced to half, the performance loss is only 4% on average. As expected, reducing the scale of each object speeds up the computation time because the amount of data per target drops. A direct relationship between the overall recognition performance and scale change along with the required processing time per trial is presented in Figure 4- 14.

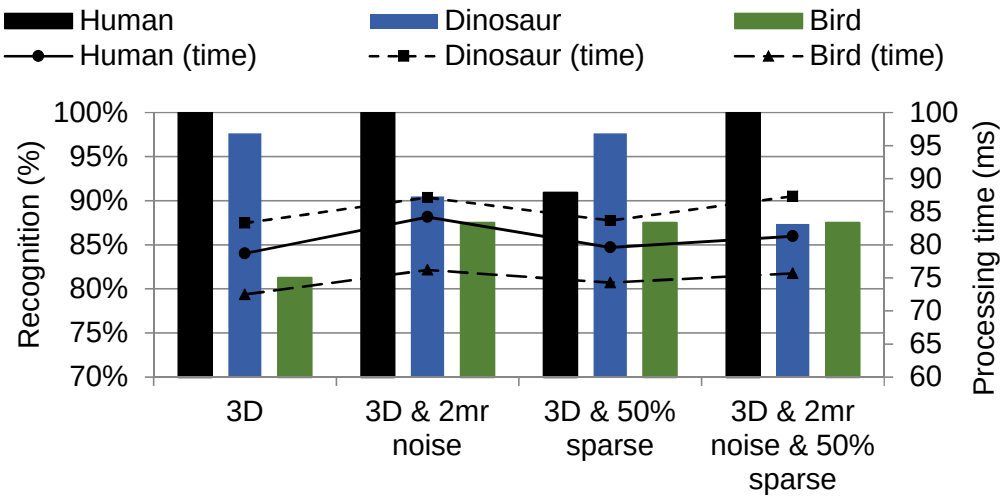


Figure 4- 13 UWA dataset inter-class recognition results at scale x1

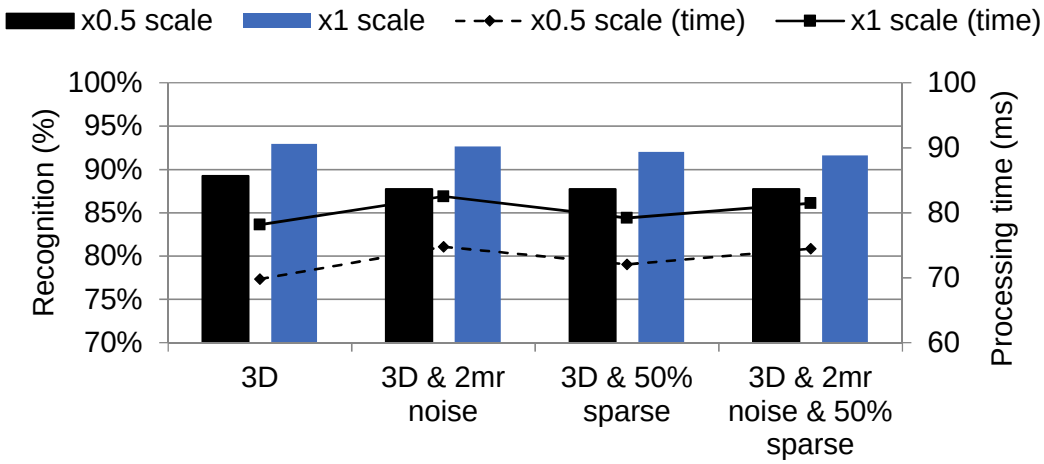


Figure 4- 14 UWA dataset inter-class average recognition results over scales x0.5 and x1

4.2.7.2 Princeton Shape Benchmark

The effectiveness of the suggested approach is further evaluated on a military target set of the Princeton Shape Benchmark. Trials are equal to Section 4.2.7 and the targets chosen are typical military representatives, namely a MBT, a helicopter and a fighter aircraft (Figure 4- 15). This database has a collection of point clouds generated from CAD models having a relatively small number of points. To provide a realistic representation of those models, points are populated with Poisson sampling [164] increasing their ideal 3D point cloud to 87,000 points

per target on average. 3D rotation is challenged with a 60° increment creating 216 viewings per target. During rotation, self-occlusion is taken into account via implementing the HPR algorithm [157].

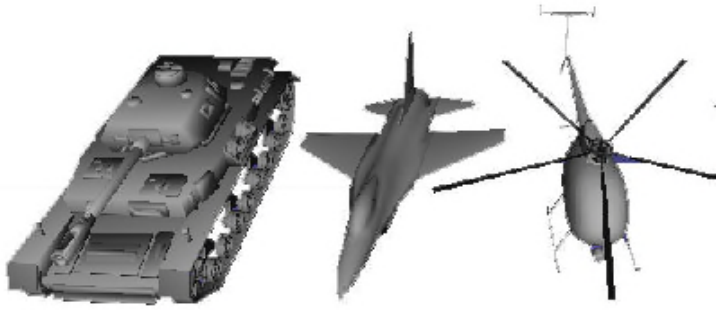


Figure 4- 15 Typical military targets from the Princeton shape benchmark, (*MBT, Fighter* and *Helicopter* classes are shown in mesh for better visualisation) (image from [13])

The first set of trials includes the cases presented in Section 4.2.7 (a) – (d). In the first trial (*3D*), the average recognition is 97.3% in 111ms. The highest performance is gained by the *Fighter* class which has a quite stable performance under all nuisances. Compared to the UWA dataset, the processing time for this dataset is increased because the targets of that dataset provide more keypoints that in turn increase the computational time of the SURF feature matching process and so of the entire recognition process. For the rest of the trials, performance is still high in the order of 96%-98% in 105ms. Detailed results per target are presented in Figure 4- 17.

The next set of experiments involves the cases previously challenged, but combined with scale change. Performance loss is only 1.5% on average at half the original scale, while processing time decreases accordingly. The relationship between scale change vs. recognition performance and processing time, per trial, is presented in Figure 4- 16.

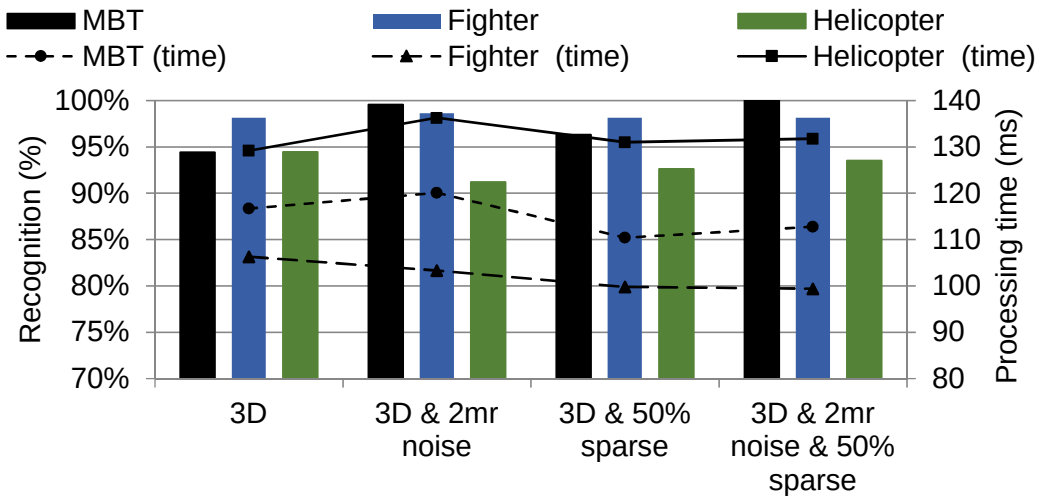


Figure 4- 17 Princeton shape benchmark military targets recognition results at scale x1 (top) and average results over scales x0.5 and x1 (bottom)

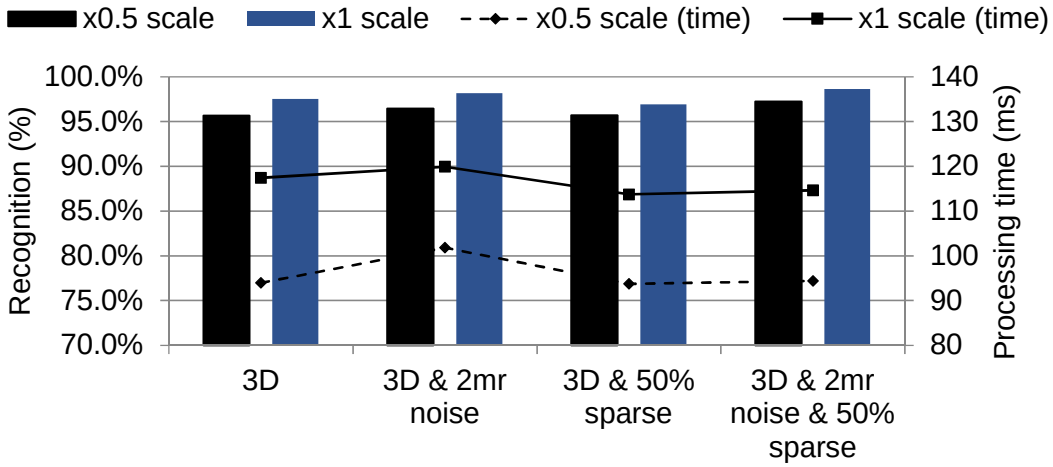


Figure 4- 16 Princeton shape benchmark military targets recognition results at scale x1 (top) and average results over scales x0.5 and x1 (bottom)

4.2.7.3 Comparison with the Rotational Projection Statistics (RoPS) algorithm

Although the PDE based ATR algorithm encodes the target in a global manner, during the following trials, it is challenged against the local based 3D descriptor RoPS [14]. The reasoning is that current global based approaches are designed to recognise objects composing of distinguishable primitive shapes, i.e. concave, convex or planar objects such as household items. In contrast to that, both databases exploited comprise of highly complex objects that are normally found in experiments related to 3D local based descriptors. Similarly to the trials of Sections 4.2.7.1 and 4.2.7.2, the single template case is considered.

The reasoning for selecting RoPS among the existing local based descriptors is because RoPS is claimed to be one of the most robust 3D descriptors currently available [90] that outperforms the Spin Image, THRIFT and SHOT [57]. During all experiments, RoPS is tuned at the parameters reported by its authors. Specifically, [61] proposes a random selection of 5000 template and 1000 target keypoints that are encoded with RoPS. Similarly to paragraph 3.2.6.4, the transformation hypothesis generation and verification process is substituted with a matching quality criterion. As a reminder, the latter considers as the correct template match the one providing the smallest average Euclidean feature distance. This modification maintains the performance of RoPS and discards the pose estimation capability, which is a time-consuming process.

The first trial regards the UWA dataset under the same parameters as Section 4.2.7 (a) – (d) but only under a scale *x1* as RoPS is scale invariant in the region of *x1-x2* [59], [167]. In all trials, the suggested descriptor consistently outperforms RoPS by achieving on average 41.7% higher recognition rate. In addition, the suggested method is two orders of magnitude faster as RoPS operates in minutes while the suggested PDE method in milliseconds. Detailed results are presented in Figure 4- 18. It is worth noting that the poor performance of RoPS in the Gaussian noise trial is due to the excessive noise added ($2\overline{mr}$) which overcomes the noise level that RoPS can handle ($0.5\overline{mr}$) [59].

The second experimental evaluation is on the same set of the Princeton shape benchmark military subset as in 4.2.7.2. The remaining experimental setup is the same as for the previous UWA dataset. Compared to RoPS, the suggested PDE descriptor has 53% higher recognition performance and is simultaneously 210 times faster to execute. The former is because RoPS cannot exploit a single template but requires several partial views of an object. As previously, the excessive noise added to the targets overcomes the invariance of the RoPS descriptor. Detailed results per trial and target can be found in Figure 4- 19.

Finally, the PDE algorithm has a notable lower descriptor storage memory demand to store the templates compared to RoPS as PDE requires 82KB/template on average, while RoPS 6400KB/template.

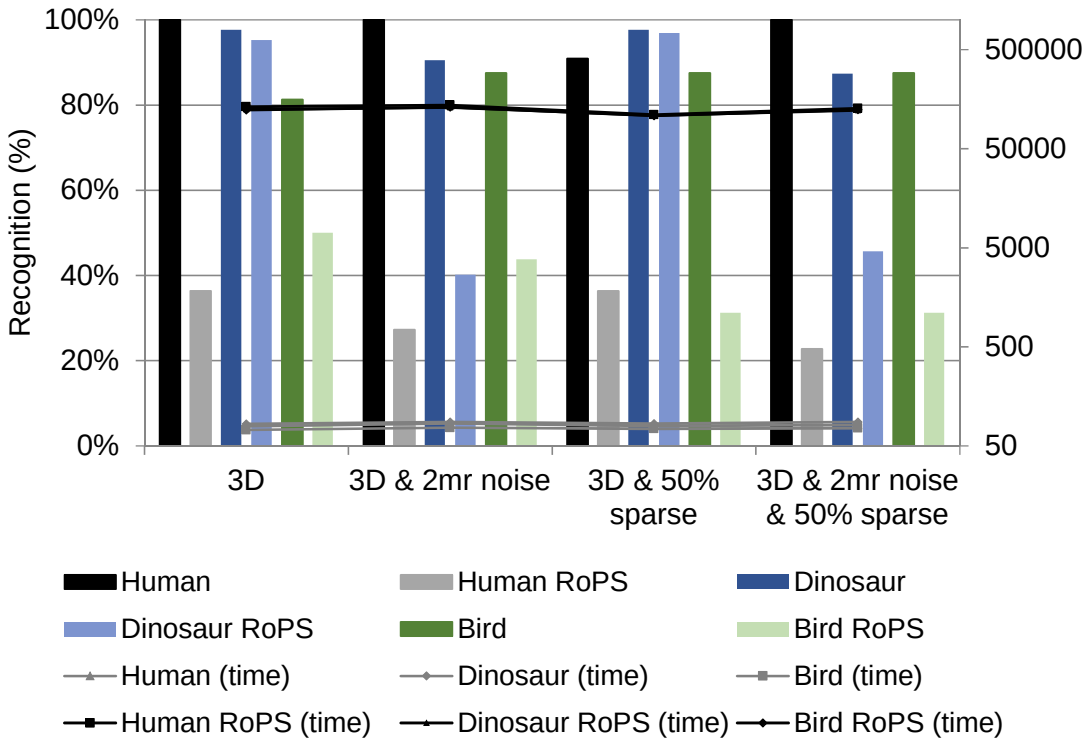


Figure 4- 18 PDE based and RoPS comparison on the UWA dataset at scale x1

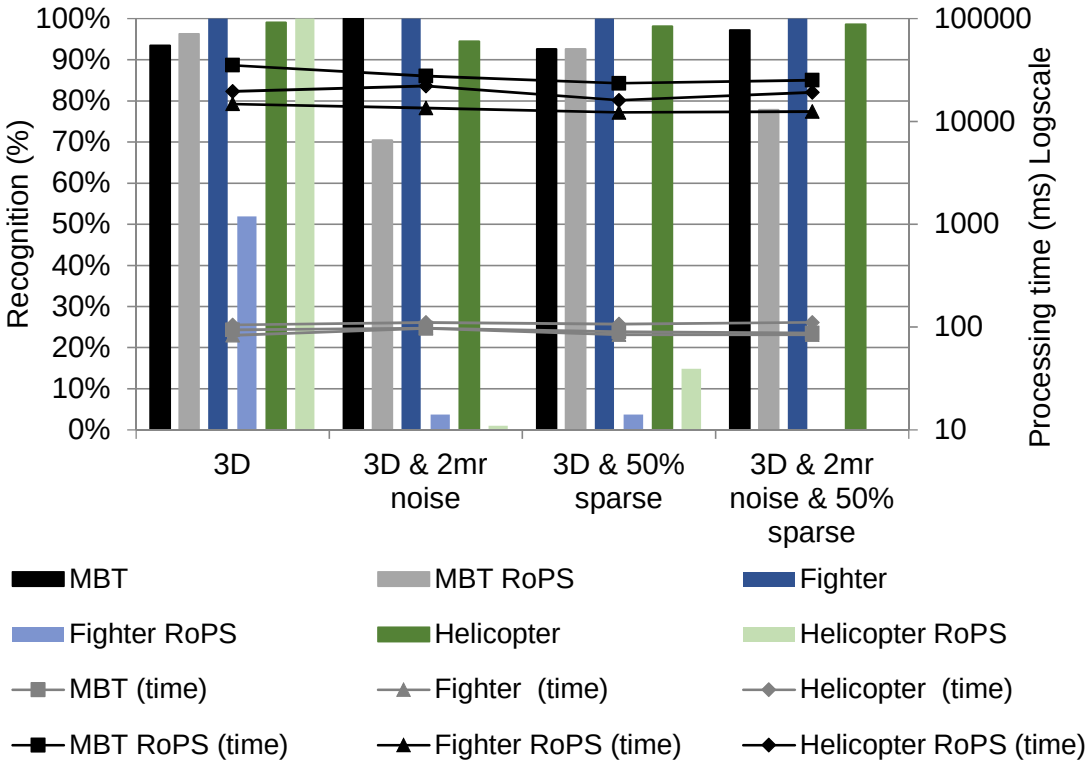


Figure 4- 19 PDE based and RoPS comparison on the Princeton shape benchmark military targets at scale x1

4.3 Conclusion

A computational efficient 3D global based ATR algorithm is proposed. The reduced template size and its computational efficiency are appealing features for object class ATR problems of clear background type scenarios or of segmented targets. In contrast to current global based descriptors, the robustness of the suggested PDE based algorithm is tested against highly complex targets instead of objects consisting of a set of primitive shapes. In addition, performance trials involve 3D target rotation, scale changes, noise and subsampling perturbations.

Due to the confidential nature of true military laser scans, trials are constrained to commercial-off-the-shelf datasets. On the UWA dataset, which comprises of real laser scanned targets, the suggested 3D descriptor gains an overall recognition of 90% in 77ms. Tested on a selection of military targets from the

Princeton shape benchmark, the suggested approach achieves on average 97% in 106ms.

Due to the high complexity of the models in the datasets, the 3D descriptor proposed is challenged against the state-of-the-art local based 3D descriptor RoPS. Trials show higher recognition rates in less time and smaller descriptor storage memory demand for the suggested descriptor.

It is worth reminding that although the suggested algorithm aims at recognising military targets, during trials the computer vision UWA dataset was also used. This was done for the following reasons: First, to ease the performance comparison with the existing computer vision 3D pattern recognition approaches by adopting standard datasets and robustness experiments e.g. invariance to standard noise levels. Second, to increase the evaluation complexity because the UWA dataset consists of real (non-simulated) non-ideal point clouds.

5 Local Based 3D ATR

Local feature based techniques describe local patches around a point of interest, i.e. support region, providing a valuable solution to partially visible objects in occluded scenes, in object registration, pose estimation and object recognition.

Although literature suggests various ways of categorising local based descriptors, they are constrained by the characteristics of the descriptor itself neglecting the origin of the data they are applied on (domain) and the possible pre-processing required. Therefore, in Section 2.4.2.4. an extended local based descriptor roadmap was suggested amending the current taxonomies proposed in [52], [57], [112] with domain and pre-processing features.

Section 2.4.2.3 and Table 2- 1 presented a great number of algorithms in the 3D local domain. It is obvious that the majority are histogram based solutions relying on a LRF, with the downside though that the robustness of these descriptors heavily relies on the repeatability and the accuracy of the LRF. For completeness, a selection based on the robustness and popularity of 3D descriptors will be analysed and discussed in the following sections.

This chapter contributes the local based 3D ATR domain with the Histogram of Distances (HoD), the HoD-Short (HoD-S), the Binary HoD (B-HoD) and the local D1 Shape Distribution (Local D1).

5.1.1 Spin Image group

5.1.1.1 Spin Image

One of the first and most cited local 3D descriptor is the Spin Image [64] which has already been used for military type ATR applications. An analysis of the Spin Image descriptor [64] has already been presented in Section 2.4.4.1.

5.1.1.2 TriSI

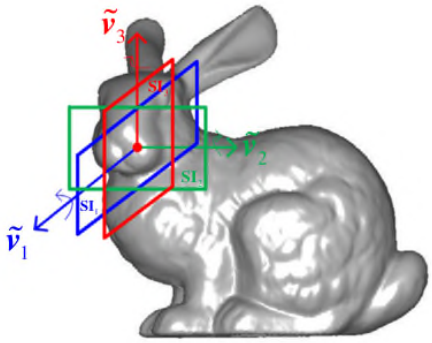
Guo *et al.* [62], [63] recently extended Spin Images by applying a Spin Image descriptor on each of the three axes of a LRF. TriSI i.e. Tri-Spin Images is the concatenation of the three Spin Images, one per axis of the LRF.

Although TriSI is highly influenced by Spin Images, it has two major differences. First, the LRF it uses is unique and robust to perturbations as it relies on the spatial point localisation information of a local surface. This is because LRF is based on eigenvalue decomposition of a weighted scatter matrix created by the vertices belonging to the support region. Sign disambiguation relies on aligning each axis to the majority of the point scatter. Second, TriSI has improved descriptiveness as it creates three signatures per support region, in contrast to the Spin Images that create only one. This triple-signature scheme as per vertex, compensates the information loss induced by a single Spin Image during the cartesian-to-cylindrical coordinate transformation procedure. Figure 5- 1 depicts the TriSI descriptor.

Although TriSI is claimed to be robust to noise and varying mesh resolutions [62], it has a number of drawbacks in relation to missile oriented applications:

a. It is prone to clutter, occlusion and suffers from keypoint localisation errors [90].

b. Trisi has an even higher computational burden compared to the already processing deficient Spin Images as it requires estimating a LRF and three Spin Images per keypoint.



c.

d. **Figure 5- 1 TriSI descriptor. Given a keypoint a LRF is established and one Spin Image per axis is estimated. Trisi is the concatenation of the three Spin Images (image from [60])**

5.1.2 Rotational Projection Statistics (RoPS) group

5.1.2.1 RoPS

Guo *et al.* suggested in [59]–[61] a 3D descriptor that is based on a local statistical analysis of a projected point cloud that is previously transformed into a mesh. Hence, given a keypoint $Pi, \{i \mid i \in \mathbb{N}\}$ acting as a centroid, a spherical volume ***V*** with radius *r* is extracted that includes the point set noted as $\boldsymbol{P_i}$. Then from $\boldsymbol{P_j}$ a weighted scatter matrix is created that is used to define the LRF of ***V***. Weights are based on the Euclidean distance of P_i to each of the point in $\boldsymbol{P_i}$. Details on the LRF construction are presented in Section 5.1.7.1 that also includes an analysis of the current LRF calculation methods. For completeness, TriSI and all RoPS variants that will be analysed shortly share the same LRF.

Given the LRF at P_i, the vertices $\boldsymbol{P_i}$ within ***V*** are rotated at a predefined angle along each axis and then are projected on the planes of the LRF. This strategy creates one distribution matrix *D* of size *LxL* per plane. Finally, each matrix *D* is encoded based on the central moments $\mu_{nm,}$ mn={11,12,21,22} and the Shannon entropy, whose concatenation forms the RoPS descriptor:

$$\mu_{mn} = \sum_{i=1}^{L}\sum_{j=1}^{L}\left(i-\bar{i}\right)^{m}\left(j-\bar{j}\right)^{n} D(i,j) \qquad \textbf{(5- 1)}$$

where

$$\bar{i} = \sum_{i=1}^{L}\sum_{j=1}^{L} iD(i,j) \text{ and } \bar{j} = \sum_{i=1}^{L}\sum_{j=1}^{L} jD(i,j) \quad \textbf{(5- 2)}$$

and

$$e = -\sum_{i=1}^{L}\sum_{j=1}^{L} D(i,j)\log\left(D(i,j)\right) \quad \textbf{(5- 3)}$$

RoPS is one of the most recent state-of-the-art 3D descriptors with enhanced robustness to noise and mesh resolution variation [21], [90]. The major drawback of RoPS, in relation to military real-time applications, is its processing deficiency that is mainly due to the complicated LRF construction [13], [21], [90]. Figure 5-2 presents a breakdown of the RoPS descriptor calculation.

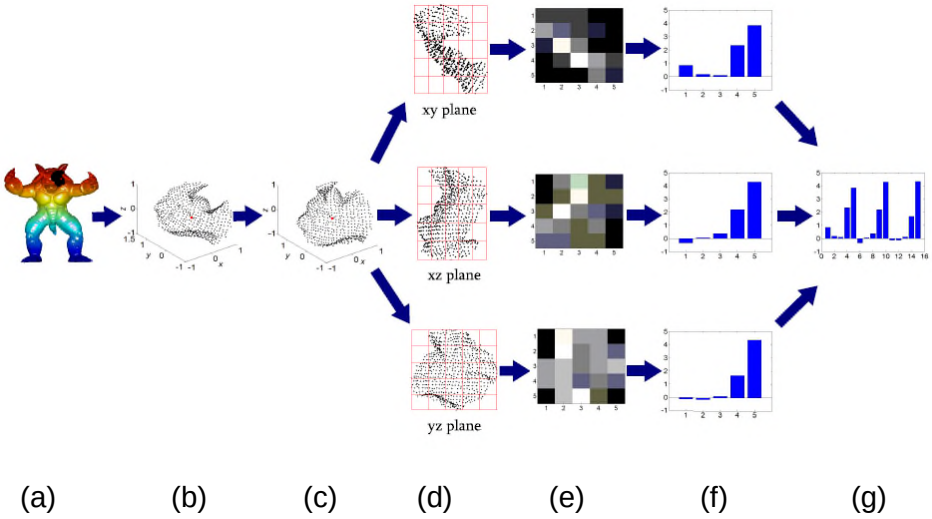


Figure 5- 2 RoPS descriptor (a) model with the local surface indicated (b) spherical support region *V* (c) LRF estimation and re-orientation of *V* (d) 2D projections of the local surface (e) 2D distribution matrix accumulating points within each bin (f) low order statistics and Shannon entropy per distribution matrix (g) final RoPS descriptor (image from [59]**)**

5.1.2.2 Colour RoPS (C-ROPS)

Guo *et al.* extended RoPS by incorporating colour information [56]. The new descriptor entitled C-RoPS encodes spatial information, as the standard RoPS, and texture information based on the *HSV* colour space. The fusion of RoPS and C-RoPS at a decision level provides enhanced performance compared to the standalone RoPS [56].

5.1.2.3 Point Cloud RoPS (PC-ROPS)

RoPS requires the object being in a mesh form that has the disadvantage of an additional processing burden to calculate the mesh itself. Therefore, Lu *et al*. proposed the PC-RoPS [45] which extends RoPS to handle directly point clouds. Beyond that modification, RoPS and PC-RoPS share the same concept and performance.

5.1.2.4 Multi Scale ROPS (MS-RoPS)

Lu *et al.* extended RoPS from a scale constrained 3D descriptor that operates in the *x1* to *x2* region [59], [167] into a multi-scale one [58]. Their solution relies on describing each keypoint with multiple RoPS descriptors of variable support region i.e. encoding radius. The MS-RoPS of the scene is then matched against all RoPS features of the templates that have a single support region. Figure 5- 3 depicts the multi scale RoPS description concept.

5.1.2.5 Improved RoPS (IRoPS)

Zeng *et al*. enhanced the discriminative power of RoPS by encapsulating depth and shape information [142]. Thus, IRoPS calculates on each distribution matrix the same low order central moments and the Shannon entropy, but in addition, it estimates the mean and the variance of the depth information of each distribution matrix. Experimental evaluation on the UWA dataset [81] has shown a clear improvement over the original RoPS descriptor.

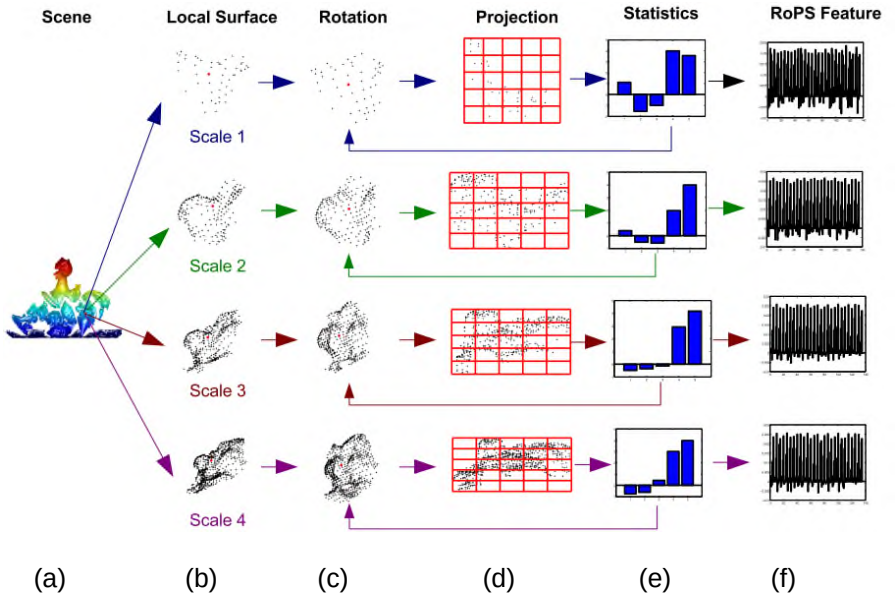


Figure 5- 3 Multi Scale RoPS descriptor (a) model with the local surface indicated (b) spherical support region extraction under multiple scales (c) LRF estimation and support region re-orientation (d) 2D distribution matrix accumulating points within each bin (e) low order statistics and Shannon entropy per distribution matrix per support region (f) final MS-RoPS descriptor comprising of multiple RoPS descriptors (image from [58]**)**

5.1.3 Signature of Histograms of Orientations (SHOT) group

5.1.3.1 SHOT

Tombari *et al.* in SHOT [52], [112] encode information about the spherical volume ***V*** of a point cloud that is centred on a keypoint $Pi, \{i \mid i \in \mathbb{N}\}$ and has a support region *r*. ***V*** is divided into eight sub-volumes along the azimuth, two along the elevation and two along the radial dimension. For every sub-volume, an 11 bin-sized 1-dimensional local histogram is computed considering the variation between the normal of the keypoint P_i and the normal of each sub-volume. Each angular variation is quadrilateral interpolated to compensate potential LRF misalignments. Finally, the histogram is normalised to sum to one such as SHOT gains robustness to point density variations. Figure 5- 4 (a) presents the SHOT descriptor.

SHOT is commonly accepted as one of the state-of-the-art descriptors [25], [62], [90] capable of achieving high quality pattern recognition performance in noisy

scenes [90] in a computationally efficient manner [21], [90]. Although SHOT is fast to execute, a GPU implementation has been suggested [193] to speedup SHOT even further.

Despite SHOT being robust to noise, when the noise level increases substantially then performance degrades faster compared to other 3D descriptors [21]. Further disadvantages involve moderate robustness to non-uniform subsampling [13], to clutter and occlusion [90], [144].

5.1.3.2 Colour-SHOT (C-SHOT)

Tombari *et al.* extended SHOT by encompassing texture information [136]. Given the same spherical subdivision as for SHOT, each vertex is associated with its corresponding *CIELab* colour triplet. Finally, the C-SHOT descriptor is the concatenation of the shape and texture related histograms. Since the latter two histograms encode information of different nature, the number of bins associated with each histogram can be of different sizes. Equally to the shape description part, the texture histogram is quadrilateral interpolated and is summed to one. Figure 5- 4 (b) presents the C-SHOT descriptor.

As expected, feature-level fusion enhances performance [194] and therefore incorporating texture information improves the recognition capability of C-SHOT compared to the original shape based SHOT [169]. Although the processing time increases and the descriptor length can be four times the original SHOT descriptor size, C-SHOT manages to balance performance with time complexity [169].

A downside of C-SHOT is being prone to illumination variation as it can affect the texture related histogram. The latter is 75% of the entire C-SHOT descriptor length and thus it is expected that illumination changes will heavily affect C-SHOT’s robustness.

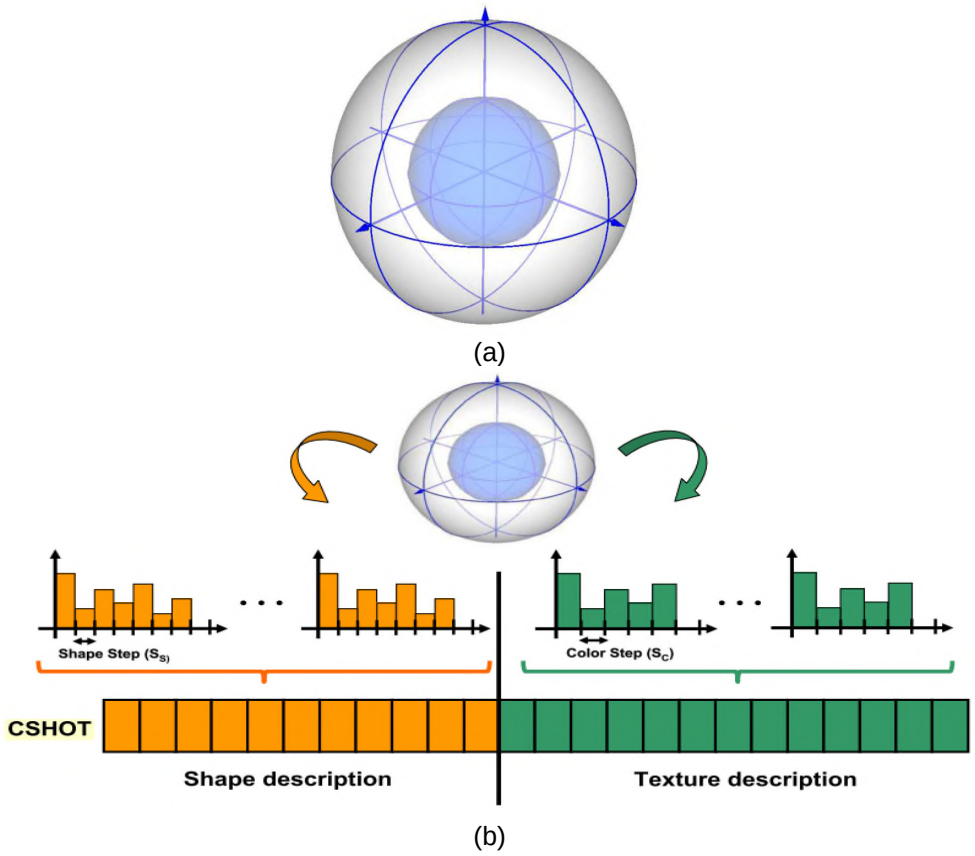


Figure 5- 4 SHOT variations as proposed by Tombari *et al.* (a) original description grid for SHOT (b) C-SHOT (image from [112]**)**

5.1.3.3 BSHOT

Prakhya *et al.* [143] further extended SHOT by transforming it from a floating point descriptor into a binary one. The transformation procedure they suggest is generic and can be applied to any descriptor. Transforming SHOT into B-SHOT is achieved by grouping the elements of the SHOT descriptor into groups of four. Each group undergoes several arithmetic trials considering all possible sum combinations within the four-element group such as summing all four elements or summing three out of four elements. Feature elements that provide sums exceeding pre-defined thresholds are substituted with the binary value of one, otherwise with zero.

The main advantages of B-SHOT are the vast reduction in descriptor storage memory requirements and the efficient feature matching process based on

Hamming distances. A disadvantage of B-SHOT is the inevitable information loss during binary quantisation.

5.1.4 3D Shape Context (3DSC) group

5.1.4.1 3DSC

Frome *et al.* [55] propose a method that sums the vertices contained in a gridded spherical volume which is centred on a keypoint. Specifically, a spherical volume ***V*** centred on the keypoint $Pi,\{i \mid i \in \mathbb{N}\}$ with radius *r* is extracted and the normal *n* of that volume is calculated. The normal *n* is calculated based on the best fitting plane to the vertices belonging to ***V***. 3DSC is LRA based with the “north pole” of the description grid overlaid to ***V*** being aligned with the normal *n*. Then ***V*** is divided into several sub-volumes along the azimuth, elevation and radial dimension. While the first two are linearly spaced, the latter is logarithmically spaced such as to enhance 3DSC’s descriptiveness. Finally, the 3DSC descriptor is created by accumulating a weighted sum for every sub-volume. Figure 5- 5 (a) depicts the 3DSC descriptor.

Although 3DSC is processing efficient [59] and robust to occlusion, clutter and to the distance of $\boldsymbol{P_i}$ to the mesh boundary [90] it has a number of limitations:

a. The LRA is not robust to azimuthal rotation. To overcome this, each keypoint P_i is multiple times described such as to cover all possible azimuthal rotations [89], [111], [112].

b. It has the largest descriptor length (1980 elements) increasing substantially the feature matching time [112].

c. It is sensitive to the mesh resolution and keypoint localisation error [90].

5.1.4.2 Unique Shape Context (USC)

The major limitation of 3DSC is its non-azimuthal rotation invariance. Therefore, Tombari *et al.* extended 3DSC by proposing USC which substitutes the LRA with a LRF [42]. The rest of the USC algorithm is identical to 3DSC. For completeness, USC and SHOT share the same robust and repeatable LRF. An analysis of current LRFs is presented section 5.1.7.1.

Although USC imposes an extra processing burden to establish the LRF, just by neglecting the multi-azimuth features description, USC is overall faster to execute and match compared to 3DSC. In addition, USC requires significantly less storage memory compared to 3DSC [57]. Further strengths of USC are its robustness to noise and occlusion [90]. Figure 5- 5 (b) presents the USC concept.

Downsides of USC are:

a. Poor robustness to varying mesh resolution and moderate invariance to clutter [51].

b. The large descriptor size increases the feature matching time prohibiting USC from time constrained applications.

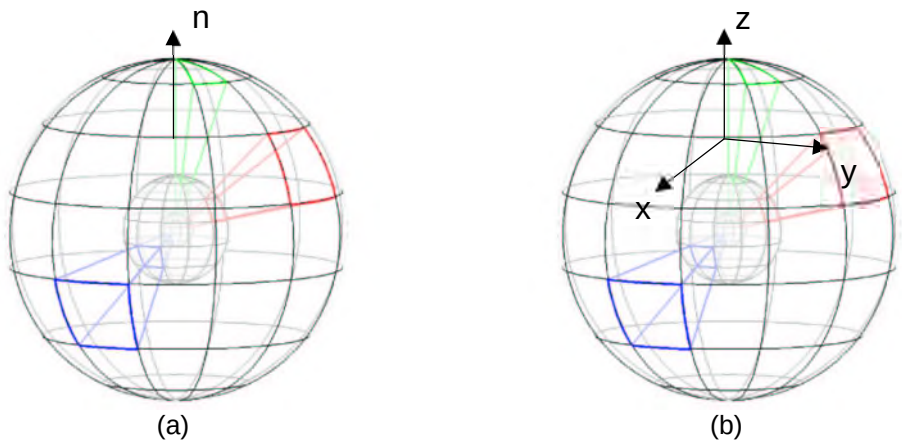


Figure 5- 5 3DSC group of descriptors (a) 3DSC (b) USC (image from [90]**)**

5.1.5 THRIFT

Flint *et al.* [65] extend the 2D SURF and 2D SIFT into the 3D domain by proposing THRIFT which exploits the keypoint detection scheme of SURF and extends the description concept of SIFT. In specific, 3D keypoints are detected by identifying local extremes of the approximated determinant of a 3D Hessian matrix on a normalised density map. The density map is created by transforming a point cloud into a voxel type representation, where each voxel is associated with the normalised number of points it contains. During the description phase, each

vertex of the point cloud is associated with a normal based on the best fitting plane aligned with the vertices allocated in a pre-defined support region. Then, for each keypoint the associated normal vector is compared against the normal of its neighbouring points and this angular difference is transformed into a 1D histogram. The latter is defined as the THRIFT descriptor.

Although THRIFT is robust to occlusion, is fast to implement [90] and does not require a LRA, it nevertheless has a number of disadvantages:

a. It is very sensitive to noise [51], [60], [65] because THRIFT relies on normal estimation that can be prone to noise.

b. Features are not very distinctive [73], [170].

c. The density function is sensitive to regions that have overlapping data [73].

d. It has a rapid performance deterioration against mesh decimation [60]. In the case of combined mesh decimation and noise, THRIFT completely fails to work [59].

e. It has a high processing burden [195] due to the multiple normal estimations and because it requires transforming the scene into a voxel type representation.

5.1.6 Point Feature Histogram (PFH) group

5.1.6.1 PFH

Rusu *et al*. [130] encode the geometrical properties of the vertices $Pi,\{i \mid i \in \mathbb{N}\}$ belonging to a sphere $\boldsymbol{V}$ of radius r centred at a keypoint P. Initially, the normal vector η_i of the P_i vertices in $\boldsymbol{V}$ is estimated (via estimating the best fitting plane to k-nearest neighbours around each P_i). Then the η_i are re-oriented to align them with the scene's viewpoint. For every point pair P_i , P_j within $\boldsymbol{V}$ having normal n_{p_i} and n_{p_j} respectively, a source point P_s and a target point P_t are selected based on the following constraint:

$$\left.\begin{matrix} P_t = P_i, P_s = P_i & if\ \left\langle n_i, p_j - p_i \right\rangle \le \left\langle n_i, p_j - p_i \right\rangle \\ P_s = P_i, P_t = P_i & elsewhere \end{matrix}\right\} \quad \textbf{(5- 4)}$$

with $i \neq j$ and $j < i$. Then, for every source – target point pair P_s - P_t a LRF is estimated with axes:

$$u = n_s, \quad v = \left(P_t, P_s\right) \times u, \quad w = u \times v \quad \textbf{(5- 5)}$$

The four features encoded by the PFH descriptor i.e. three angular and one distance are given by:

$$\left.\begin{matrix} f_1 = \left\langle u, n_t \right\rangle \\ f_2 = \left\| p_t, p_s \right\| \\ f_3 = \dfrac{\left\langle u, p_t - p_s \right\rangle}{\left\| p_t, p_s \right\|} \\ f_4 = a\tan\left(\left\langle w, n_t \right\rangle, \left\langle u, n_t \right\rangle\right) \end{matrix}\right\} bin\ index = \sum_{i=1}^{i \le 4} step(s_i, f_i) \cdot 2^{i-1} \quad \textbf{(5- 6)}$$

where n_t is the surface normal at P_t, s and f the source and the target point PFH features respectively and

$$step(s,f) = \left\{\begin{matrix} 0 & if\ f < s \\ 1 & elsewhere \end{matrix}\right\} \quad \textbf{(5- 7)}$$

Finally, the PFH descriptor is generated by accumulating these features into a histogram of a predefined bin size. In a later manuscript [117] the authors neglect the distance based description to enhance robustness to point density variations.

PFH is a major contribution in 3D ATR as it is the core concept of various global and local based descriptors. In the global domain PFH is the basis for VFH, CVFH, OUR-CVFH and Compressed CVFH as already presented in Chapter 4, while in the local domain, PFH has been extended to a faster variant the FPFH.

Although PFH is robust in varying mesh resolution and keypoint localisation errors [165], it has a number of drawbacks:

a. It has poor robustness to noise, clutter and occlusion [90].

b. The processing time required prevents it being exploited in real time applications [117].

Recently, Alexandru [196] proposed a multimodal PFH feature by fusing into PFH texture information. This suggestion increased the performance substantially [169] with the downside of doubling the descriptor size and increasing further the execution time of the already processing demanding PFH.

5.1.6.2 Fast Point Feature Histogram (FPFH)

Rusu *et al.* in [117] propose a faster variant of PFH, namely the FPFH, which is claimed to retain the descriptiveness of PFH. In contrast to PFH which considers all the interconnections of the vertices belonging within the extracted spherical volume ***V*** that is centred at $Pi,\{i \mid i \in \mathbb{N}\}$, FPFH exploits only the immediate to the keypoint k-neighbours and the immediate neighbours of each of the k-neighbours, namely the P_k, $\{k \mid k \in \mathbb{N}\}$ vertices (Figure 5- 6).

Given the reduced interconnections, the Simplified PFH (SPFH) is estimated in the same manner as PFH but considering far less vertices. Finally, for a query keypoint P_i the FPFH descriptor is given by:

$$FPFH(P_i) = SPFH(P_i) + \frac{1}{k}\sum_{i=1}^{k}\frac{1}{\omega_k}SPFH(P_k) \qquad \textbf{(5- 8)}$$

Although originally $\omega = \|P_i - P_k\|$ the authors of FPFH in a latter manuscript [191] propose $\omega = \sqrt{\exp\|P_i - P_k\|}$ for better performance.

FPFH has one of the shortest descriptor lengths with only 33 elements, and therefore FPFH feature matching is extremely efficient. Beyond that, FPFH is quite robust under uniform point cloud decimation [90], [165]. The drawbacks of FPFH include being prone to noise [112], clutter and occlusion [90].

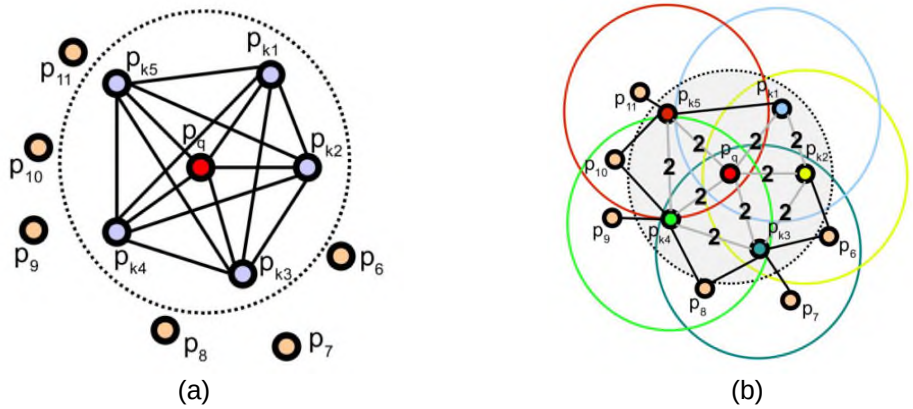


Figure 5- 6 PFH group of descriptors showing the point-pair interconnections for (a) PFH (b) FPFH (image from [117]**)**

5.1.7 Discussing Current Local Based 3D descriptors

Although several algorithms are available (presented in Table 2-1), most of them require an accurate and robust LRF/A on which the descriptor is defined. Accuracy and robustness of the LRF/A highly depend on the complexity of the LRF/A algorithm, which in turn directly impacts the total processing time. In addition, it is very challenging to define a repeatable LRF/A under noise and/or point cloud density variation [89], [111].

On the other hand, 3D ATR solutions that do not require a LRF/A have their own individual deficiencies. In specific, THRIFT has a high processing burden [195] while Heat Kernel Signature (HKS) [132] requires a massive amount of RAM which is in the order of 6GB for a point cloud of 30,000 vertices [87]. Invariant Shape Contexts (ISC) [138] and Hidden Markov Model (HMM) [54] are applied on the 3D data mesh but 3D sensors do not provide the interconnectivity of the vertices and therefore extra processing is demanded that increases the total computational time. Concerning the Variable Dimensional Local Shape Descriptors (VD-LSD) [128], the process of selecting an optimized subset of VD-LSDs is a time consuming procedure [59].

Based on those facts, this research focuses on solutions that neglect the LRF/A requirement and via this strategy aim at reducing dramatically the processing time such as to meet the processing requirements of missile seeker applications.

Beyond the computational efficiency required, the proposed 3D Local descriptor should achieve high recognition performance even under the combination of severe noise and non-uniform subsampling of the point cloud.

5.1.7.1 Computational Cost for LRF/A Establishment

This section investigates the percentage of processing time spent to establish the LRF/A as a fraction over the entire processing time required to create the descriptor. Investigation involves three LRF's and one LRA which are: the ones used by SHOT [112] and RoPS [60] that are two of the most recent and robust LRF's [111], 3D Tensor's LRF [126] and the LRA used by the Spin Images [197]. These LRF/A are estimated by:

a. 3D Tensor's LRF: Given a keypoint $P_c, \{c \mid c \in \mathbb{N}\}$, a spherical volume $\boldsymbol{V}$ with radius r is extracted that encompasses k vertices $P_i, \{i \mid i \in \mathbb{N}, i \leq k\}$. Unit vectors of the LRF are given by eigenvalue decomposition of the normalised vectors of the covariance matrix:

$$C = \frac{1}{k}\sum_{i=1}^{k}\left(P_i - \hat{P}\right)\left(P_i - \hat{P}\right)^T \quad \textbf{(5- 9)}$$

where $\hat{P}$ is the barycentre of $\boldsymbol{V}$,

$$\hat{P} = \frac{1}{k}\sum_{i=1}^{k} P_i \quad \textbf{(5- 10)}$$

b. SHOT's LRF: Compared to the Tensor's LRF estimation this one has two major differences. First, the barycentre $\hat{P}$ in the covariance matrix is substituted with the keypoint P_i. Second, the covariance matrix is weighted to enhance robustness to clutter:

$$C = \frac{1}{\sum_{i:d_i \leq R}(R - d_i)}\sum_{i=1}^{k}(R - d_i)(P_i - P_c)(P_i - P_c)^T \quad \textbf{(5- 11)}$$

with $d_i = \left\| P_i - P_c \right\|_2$ and $R=max(d_i)$.

c. RoPS' LRF: For a keypoint P belonging to the i^{th} out of N triangles of a triangular mesh, with surrounding vertices P_{i1}, P_{i2} and P_{i3}, the weighted covariance matrix used to establish the LRF is given by:

$$C = \sum_{i=1}^{N} w_{i1} w_{i2} C_i \qquad \textbf{(5- 12)}$$

where

$$w_{i1} = \frac{\left| \left(P_{i2} - P_{i1}\right) \times \left(P_{i3} - P_{i1}\right) \right|}{\sum_{i=1}^{N} \left| \left(P_{i2} - P_{i1}\right) \times \left(P_{i3} - P_{i1}\right) \right|}, \quad w_{i2} = \left(r - \left| P - \frac{P_{i1} + P_{i2} + P_{i3}}{3} \right| \right)^2 \qquad \textbf{(5- 13)}$$

with $\times$ being the cross product, r the support radius and

$$C_i = \frac{1}{12} \sum_{j=1}^{3} \sum_{k=1}^{3} \left(P_{ij} - P\right)\left(P_{ik} - P\right)^T + \frac{1}{12} \sum_{j=1}^{3} \left(P_{ij} - P\right)\left(P_{ij} - P\right)^T \qquad \textbf{(5- 14)}$$

d. Spin Image's LRA: Given a triangular mesh, a triangle formed by the vertices P_1, P_2 and P_3 has a normal given by:

$$\vec{n} = \left(P_2 - P_1\right) x \left(P_3 - P_2\right) \qquad \textbf{(5- 15)}$$

In any case, a great portion of the total processing time is spent for the LRF/A creation. This timing includes the point cloud to mesh conversion where it is needed. In contrast to [111], this evaluation does not investigate the absolute overall processing time but focuses on the relative time spent compared to the total processing time needed to create the entire descriptor. This methodology is important as it directly reveals the computational gain by discarding the LRF/LRA and is independent of the hardware capabilities of the evaluation platform.

Trials are performed on the Stanford 3D scanning repository [198] and ascertain the results shown in Figure 5- 7. This trial reveals that SHOT devotes 70% of its total processing time for the LRF construction while the 3D Tensor LRF requires

66%. Compared to the SHOT's LRF, the Tensor's is faster to estimate due to neglecting the distance related weight calculation. In the case of the RoPS' LRF, the percentage increases to 88% due to its high complexity encompassing information and creating weights from the surrounding vertices. Even in the case of the relatively simple LRA estimation, the processing burden is still 50% of the total time.

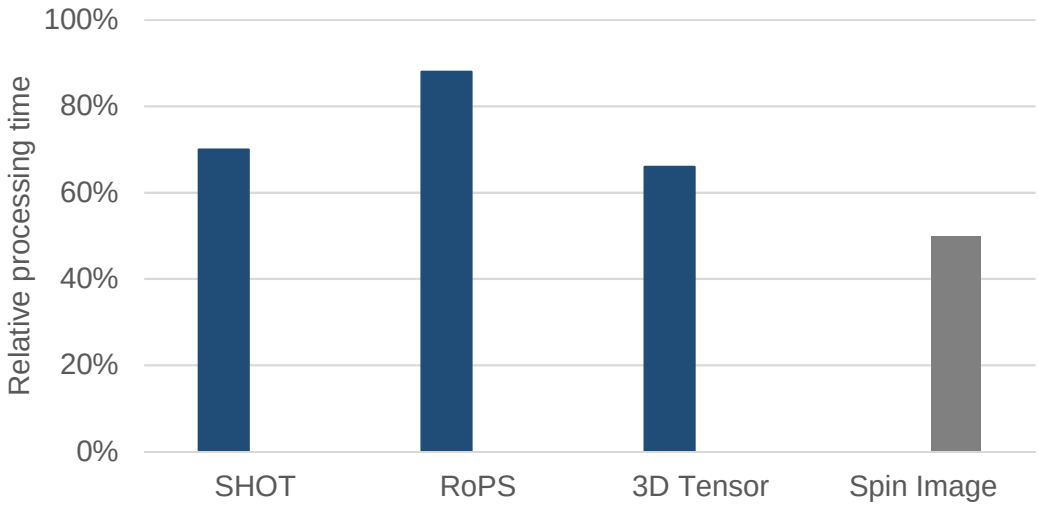


Figure 5- 7 Processing time per LRF in blue and LRA in grey

It should be noted that during this investigation only the processing burden is considered and not the performance of each LRF/A in terms of stability and robustness to perturbations such as noise and resolution.

From a solely processing time perspective, it is concluded that if these highly computational expensive LRF/A procedures were neglected and substituted with a descriptive, robust and fast to execute descriptor, this would provide an appealing solution for real-time 3D pattern recognition applications such as 3D missile ATR.

5.1.8 Conclusion based on current Local 3D descriptors

The computer vision community has already achieved high quality 3D object recognition under various occlusion and noise levels. What is still challenging is

further enhancing robustness to various perturbations such as noise and subsampling and transformations like scale change, further improving recognition performance and reducing the processing time required. The latter in specific is one of the major constraints to applying current computer vision algorithms for time-critical applications as the future missile ATR on that is discussed in this research.

5.2 Histogram of Distances for Local Surface Description

Driven by these requirements, the contribution of this section is a 3D Local descriptor named Histogram of Distances (HoD) [21] that according to the taxonomy suggested in Chapter 2 can be considered as a raw point cloud SDH descriptor. HoD aims at revealing the underlying local point-pair distance distribution and compressing it into bins. In contrast to existing Histogram based descriptors, HoD does not require any LRF or LRA estimation. Neglecting an LRF/A is an important advantage both in terms of processing efficiency as well as in robustness to high level perturbations.

HoD is inspired by Osada's D1 Shape Distribution [105] but properly modified to facilitate the descriptiveness and robustness required for a missile ATR algorithm. As a reminder, the D1 is a global descriptor and encodes the pairwise Euclidean distance distributions between the point cloud's centroid and each vertex belonging to that point cloud (Chapter 4.1.1).

The proposed HoD descriptor has the following features:

a. Simplicity: HoD is a simple and processing efficient but descriptive 3D local descriptor that encodes the local pairwise distance distributions.

b. LRF/A independence: Opposing to the vast majority of the existing local based approaches [25], [30], [37], [39], [42], [45], [50], [52], [55], [56], [58], [59], [61], [63], [64], [81], [85], [86], [89], [112], [117], [119]–[127], [129]–[131], [133]–[137], [139]–[141], [143] the necessity of a LRF/A is neglected. Therefore, external disturbances like noise and subsampling influence less the HoD descriptor and thus the recognition performance.

Additionally, neglecting the LRF/A reduces the processing time requirements.

c. Multi-layered description: HoD encompasses a dynamically changing keypoint description scheme, combined with a multi-description and multi-feature matching policy. This novel strategy enhances the recognition performance under severe noise levels, varying point cloud resolution and quality.

d. Directly applicable to the point cloud: In contrast to a considerable number of existing 3D pattern recognition solutions [37], [39], [42], [50], [52], [54], [56], [58], [59], [61], [63], [64], [81], [89], [112], [119]–[127], [129], [131]–[134], [136], [138], [140] the proposed descriptor is not applied to a mesh or a voxel representation but directly to the raw point cloud data. This manipulation saves much computational time.

5.2.1 Establishing the HoD Feature Descriptor

A point cloud $\boldsymbol{P}$ consists of the vertices $P_i=(x_i,y_i,z_i)^T$, $\{i \mid i \in \mathbb{N}, i \leq K\}$ where K is the total number of points. For each P_i acting as a centroid, a spherical volume $\boldsymbol{V}$ with support radius r is extracted that encloses the points $P_j, \{j \mid j \in \mathbb{N}, j \leq i\}$. In contrast to [81],[64],[117],[63],[100],[112],[60] where $\boldsymbol{V}$ has a fixed radius r equal to a multiple of the average template mesh resolution $(\overline{mr})$, HoD takes advantage of the average point cloud resolution per scene (mr)[3]. Although mr estimation and template keypoint description via the HoD are also done during the online phase, this strategy provides a considerable advantage. The support radius is dynamically changing depending on the severity of the perturbations i.e. noise and subsampling levels that affect the scene. Hence, in contrast to current 3D recognition approaches, HoD has a variable support region that is directly linked to the characteristics of each individual scene. An additional advantage of the variable support region is its robustness to scale changes which is a unique feature for a 3D descriptor. Current local feature based 3D descriptors have

[3] To be uniform to the current average mesh resolution notation $\overline{mr}$ explicitly used in the literature, the average point cloud resolution used by HoD is presented with the notation mr.

limited scale invariance e.g. RoPS is up to x2 [59], [167], and they rely their scale invariance on the characteristic scale that the keypoint to be described is detected in during the keypoint detection process [199]. The characteristic scale defines the description radius of the 3D descriptor for each keypoint.

Keypoints can be selected either by applying the existing 3D keypoint detectors [87] or randomly. In the following trials, keypoints are randomly selected to avoid influencing the overall performance by the keypoint detector [52].

For the description of each spherical volume ***V***, one border point is selected that acts as a local reference point P_r. In the following experiments P_r is set as the one closest to the origin of the global reference frame which is set at the sensor.

Given a reference point P_r, all pairwise L2-norms with the vertices P_j belonging to ***V*** are calculated:

$$d_j = \left\| P_r - P_j \right\|_2 \qquad \textbf{(5- 16)}$$

The d_j calculated are in the form of continuous variables and thus are highly prone to even minor spatial perturbations and missing vertices. Therefore, d_j is discretized by using the equal width interval binning method [191]. This method is fast to execute and sorts the observed continuous values d_j into B equally sized bins of width δ hence, the discretized L2-norms are given by:

$$\left\{ d_j^{'} = \left\lfloor \frac{d_j}{\delta} \right\rfloor : j \in \left\{0, \max\left(d_j\right)\right\} \ \wedge \ \delta = \frac{\max\left(d_j\right)}{B} \right\} \qquad \textbf{(5- 17)}$$

Finally, the proposed descriptor D encodes the probability mass density of the local distance distributions $d_j^{'}$ in a histogram form. This is done by encrypting counters of $d_j^{'}$ into histograms:

$$D = P_r\left(\left\{ d_j^{'}(s) = x \right\}\right), \ \left\{ x, s \,\middle|\, x, s \in \{1, B\} \right\} \qquad \textbf{(5- 18)}$$

The advantage of the histogram based encoding is enhanced robustness to nuisances that is achieved by compressing information into bins [52]. Since the

majority of the local histogram descriptors rely on a LRF/A [52], [55], [57], [59], [63], [64], [85], [117], [126], [129], [132], [136], HoD can be defined as a hybrid histogram descriptor. Furthermore, since *D* is based on the probability mass density, each descriptor sums up to one. The latter offers enhanced robustness to point cloud resolution changes [52], [59].

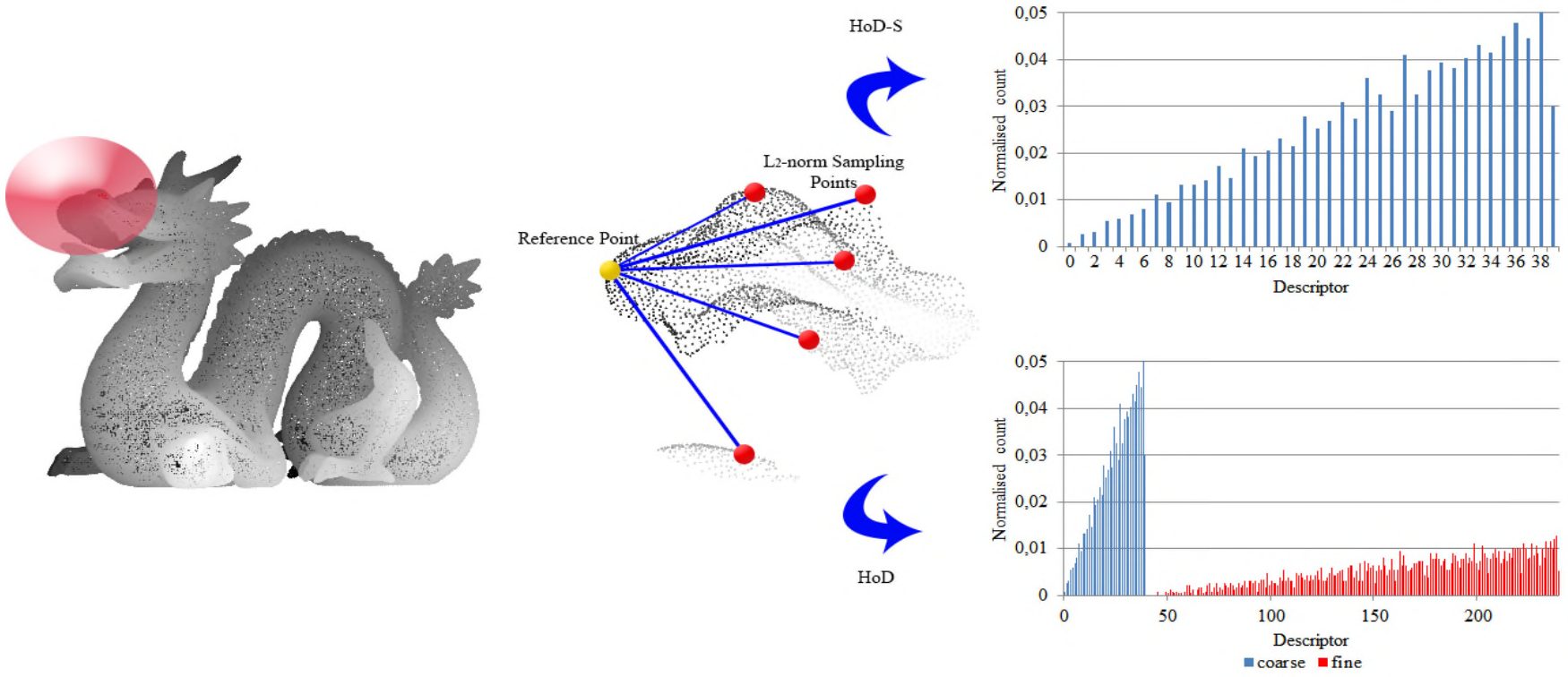
In this chapter, two variants of *D* are proposed, one that favours descriptiveness with a 240-element long descriptor named Histogram of Distances (HoD) and a shorter one with only 40 elements named HoD-S that favours processing time with a minor influence to recognition performance. The HoD, although slower than HoD-S to estimate it is still faster compared to current state-of-the-art descriptors.

A further improvement of HoD includes establishing a dual-layered bin-size distribution scheme. Instead of exploiting a single 240 element long descriptor, the HoD descriptor is split into a coarse and a fine encryption process. Initially HoD describes the spherical volume $\boldsymbol{V}$ centred at a keypoint P_i with a 40-element long descriptor in the same manner as HoD-S does. Additionally, HoD re-describes the same local patch with a finer bin distribution profile, which is 200-element long. The final HoD descriptor comprises of the concatenation of the coarse and the fine description process. Figure 5- 8 presents the suggested HoD and HoD-S descriptors.

This dual-layered bin size strategy has the advantage of enhancing the HoD's effectiveness by efficiently encapsulating the 3D structure of a high, medium or even low quality point cloud. Consequently, during the feature matching stage, the feature matches per description depth level are estimated. This multi-encoding and multi-feature matching policy strengthens the descriptiveness and the robustness of HoD to perturbations such as noise and subsampling.

5.2.2 Evaluation Process

The HoD and HoD-S descriptors are assessed with the popular 1-Precision vs. Recall curve (PR) [63],[65],[100], [112],[60],[151],[200]. For the HoD case, the PR curve is based on a set of model features $\boldsymbol{f}_i^M = \boldsymbol{f}_i^M{}_{coarse} \,||\, \boldsymbol{f}_i^M{}_{fine}$, a ground truth



(a) (b) (c)

Figure 5- 8 Histogram of Distances (HoD) concept. (a) A spherical volume *V* centred at P_j is extracted. (b) A random border point from the local area is selected as reference point (yellow) and the reference point to vertices L2-norms are calculated (in red as example). (c) L2-norms are encoded into a Histogram of Distances

transformation and the corresponding scene features $f_j^S = f_{j\ coarse}^S \,||\, f_{j\ fine}^S$. A scene feature is matched with all model features based on their x^2 distance and the NNDR criterion with a variable threshold τ. Feature matching is performed on each description level i.e. separately for the coarse and for the fine description. If the ratio of the nearest model feature $f_{i\ coarse}^M$ with the second nearest $f_{i'\ coarse}^M$ is less than a threshold τ, then the scene feature $f_{j\ coarse}^S$ and the model feature $f_{i\ coarse}^M$ are considered as a match. In the same way feature matches are established for the fine description scheme exploiting the same threshold τ. The description scheme i.e., fine vs. coarse that provides most matches is considered as the accepted domain in which recognition will be based. For the HoD-S case the matching strategy is equivalent and restricted to a single coarse feature matching strategy. Furthermore, if the Euclidean distance of the physical location of the matched keypoints is less than half the HoD's descriptor support radius, then the match is considered as *True Positive* (TP), otherwise, as *False Positive* (FP) [114]. Correspondences are established in the same manner. Recall and 1-precision are defined as [114]:

$$recall = \frac{\#TP}{\#correspondences} \qquad \textbf{(5- 19)}$$

$$1 - precision = \frac{\#FP}{\#matches} \qquad \textbf{(5- 20)}$$

where # denotes the number of the entity that follows i.e. number of TPs.

By altering the NNDR threshold values τ in the range [0,1] the PR curve is obtained which ideally would be at the upper left corner i.e., both recall and precision are equal to one.

5.2.3 HoD Parameter Setup

The descriptiveness of HoD and HoD-S depends on the number of distribution bins *B*, the support radius *r*, and the feature match metric. Therefore the performance of HoD is tested against different settings of these parameters on a

tuning dataset which comprises of 10 scenes similar to the Bologna dataset [112]. These scenes are non-uniformly down-sampled to ½ their mesh resolution and Gaussian noise is added with a standard deviation of 10% the average mesh resolution $\overline{mr}$ [52], [59]. The performance criterion is the PR curve while in parallel processing efficiency is considered with equal importance. The optimum parameter setup chosen is then applied on both HoD variants during the experimental stage.

5.2.3.1 The Number of Distribution Bins

The number of distribution bins *B* is an important parameter as it determines the robustness and the descriptiveness of the HoD descriptor. A sparse distribution scheme is more robust to perturbations such as noise and subsampling but is less descriptive compared to its dense counterpart. Thus, it is crucial to select carefully the distribution bins that balance robustness and descriptiveness.

The following trial evaluates HOD's performance on the tuning dataset with respect to the number of distribution bins, while the support radius and the feature match metric are set to 40 and x^2 respectively. Detailed results are presented in Figure 5- 9 (a) – (b). The graphs clearly show that for the tuning dataset the performance of HoD increases as the distribution bins increase. This is because as *B* increases, the descriptor encodes the local point cloud patch in greater detail. It should be noted though that as the severity of the disturbances increases i.e. higher noise and subsampling levels, the bin width δ has to decrease accordingly such as to compensate the new point-pair distance distribution that is affected by the new spatial location of the vertices within the corrupted scene. Therefore, a suitable *B* size has to be carefully selected to balance the descriptor's descriptiveness and invariance. Regarding the computational efficiency, the processing time range is 1.6s between the smallest and the largest *B* size investigated.

Considering the equal importance of recognition performance and computational efficiency, the distribution bins are set to 40 for the HoD-S and 240 for the HoD (as a concatenation of 40 and 200 bins).

5.2.3.2 The Support Radius

The support radius *r* defines the spherical volume ***V*** that the descriptor must encode. Increasing *r* enhances the descriptor's descriptiveness because the number of points P_j to be encoded increase and thus the uniqueness of the pointwise distance distribution within ***V*** is more evident. On the other hand, robustness to occlusion and clutter reduce as these interfere with the larger volume ***V*** and affect the contained vertices P_j.

During the HoD's parameter setup, several values for *r* are tested, while maintaining the bin size *B* and fixing the feature match metric. Figure 5- 9 (c) depicts the PR curves with corresponding processing efficiency. As the support radius increases, both recall and precision improve. Although performance progressively increases, significant improvement is observed from the 10mr to the 20mr radius. This is because a 10mr support radius is too small to encapsulate discriminating distance distribution information on the local point cloud within ***V***. In terms of processing time, all evaluated *r* have a time range of 1.7s. For the rest of this chapter, the support radius is set at 40mr. That radius provides to HoD high quality performance with a recall greater than 85%.

5.2.3.3 The Feature Matching Metric

Another important parameter is the distance metric employed to match the scene and the template features. Literature suggests several metrics with the most common ones being:

a. L1-norm or Manhattan, which measures the absolute value distance:

$$d = \left\| f_i^M - f_j^S \right\| \qquad \textbf{(5- 21)}$$

where f_i^M and f_j^S are the feature model and scene with index *i* and *j* with respect to $i, j \in \mathbb{N}$.

b. L2-norm or Euclidean, which measures the shortest distance:

$$d = \left\| f_i^M - f_j^S \right\|_2 \qquad \textbf{(5- 22)}$$

c. Kullback - Leibler (KL) divergence, which measures the similarity by calculating the relative entropy:

$$d = \sum\left(\left(f_i^M - f_j^S\right)\log\left(\frac{f_i^M}{f_j^S}\right)\right) \quad \textbf{(5- 23)}$$

d. Shannon Entropy (SE), which measures the disorder of the matched features:

$$d = \sum\left(-\left(f_i^M - f_j^S\right)\log\left(f_i^M - f_j^S\right)\right) \quad \textbf{(5- 24)}$$

e. x^2 distance, which measures the underlying distance of the features and emphasizes their dissimilarity:

$$d = \sum\left(\frac{f_i^M - \frac{f_i^M + f_j^S}{2}}{\frac{f_i^M + f_j^S}{2}}\right) \quad \textbf{(5- 25)}$$

HoD's performance is evaluated on the tuning dataset with respect to the feature matching metric while the support radius *r* and number of distribution bins *B* are set to 40mr and 240 respectively. Experimental results show that all metrics perform equally well with the Kullback – Leibler and the x^2 metrics presenting the lowest and the highest performance respectively. PR curves and processing efficiency are shown in Figure 5- 9 (d).

Regarding the processing efficiency, the processing time ranges by only 0.1s and thus the feature match metric is chosen purely based on performance criteria. Thus, both HoD variants are based on a x^2 metric.

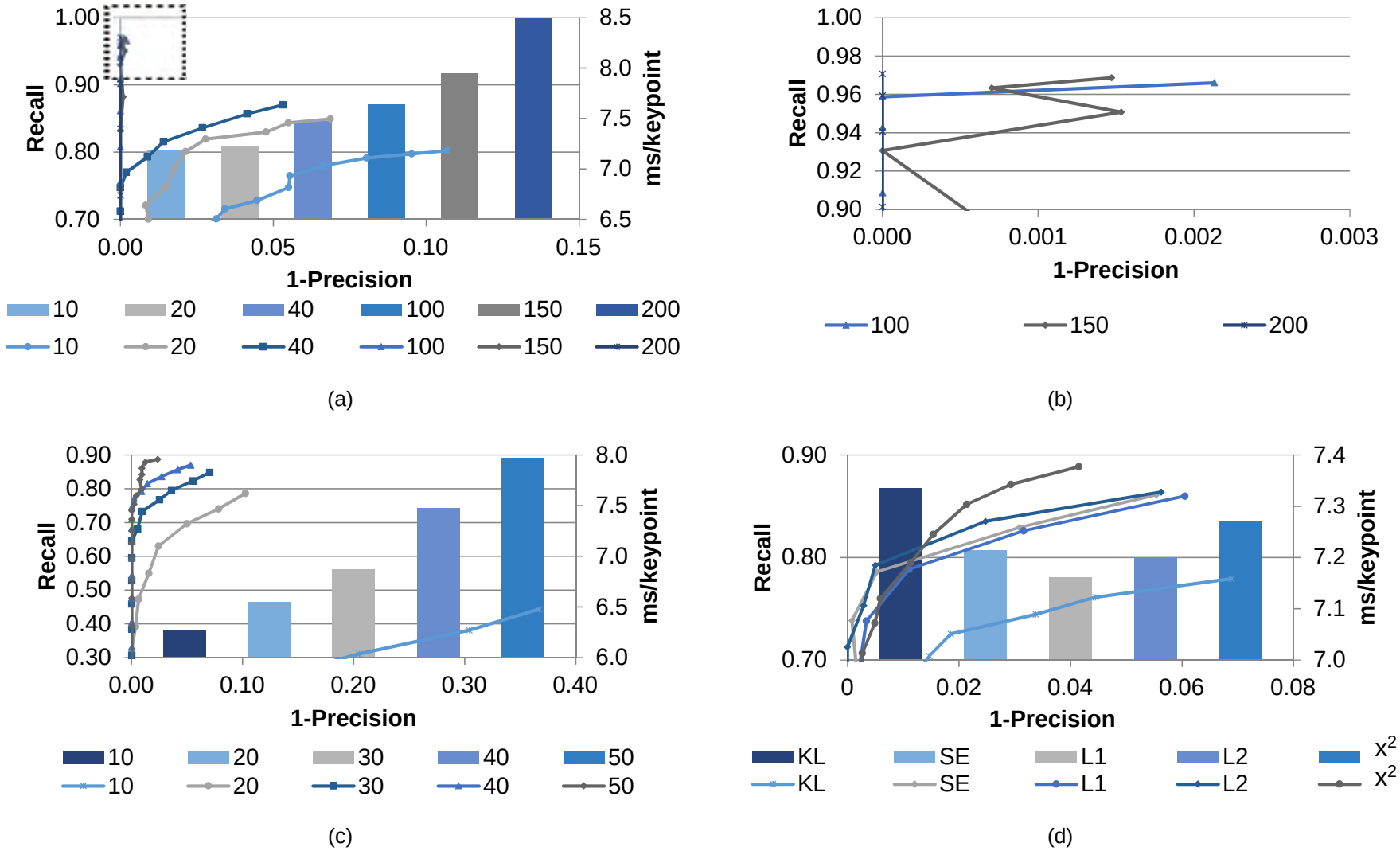


Figure 5- 9 HoD parameter setup. (a) Effect of altering the number of distribution bins (b) Magnification of the high performing region (c) Support radius as multiple of *mr* (d) Effect of the point-pair distance match metric (line shows recognition performance while bars processing time)

5.2.4 Experimental Results

5.2.4.1 Comparison with state-of-the-art approaches

HoD and HoD-S (coarse-fine and coarse only description scheme respectively) are challenged against the current state-of-the art 3D pattern recognition algorithms RoPS, SHOT, FPFH, 3DSC and the D1 Shape distribution. The latter is extended into the local domain i.e. Local D1, by sharing the same parameters as HoD while the reference point is the centroid P_c as originally proposed in [105].

Performance is assessed based on the PR curves with the support radius of the competitor descriptors independently tuned for optimal performance. Support radius tuning is done on the tuning dataset presented in Section 5.2.3 and all trials are performed in MATLAB and in C++. Implementations in C++ are obtained from the Point Cloud Library (PCL) Version 1.7.2 [200] while RoPS from MATLAB File Exchange [201]. Beyond the support radius which is tuned for the best PR curve, the rest of the parameters including the feature match metric are fixed either to the ones originally proposed by their authors or to their PCL implementation [90]. The descriptors evaluated and their parameter settings are presented in Table 5-1.

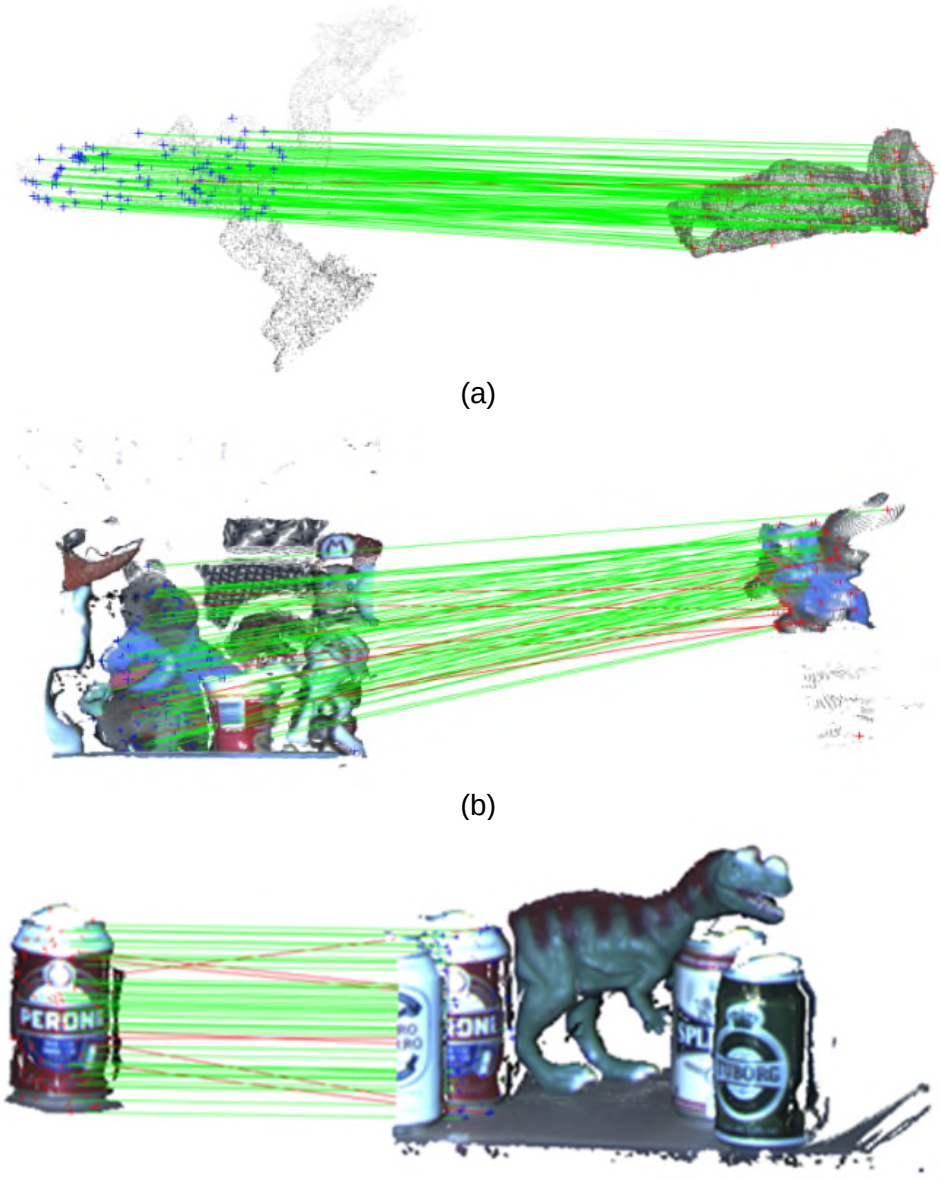
During the tuning process of FPFH its performance peaked at a support radius of $20\overline{mr}$ which is substantially smaller compared to the radius of the other descriptors evaluated. This confirms [90] which states that FPFH performance peaks at some support radius and beyond that its performance drops.

Since the ultimate implementation concerns time-critical applications, 100 keypoints from each model are selected and their corresponding ones in the scene are extracted based on their *a priori* known ground truth transformation. Random keypoint selection is preferred against exploiting a keypoint detector as errors of the detector can affect the descriptor [59].

Table 5- 1 Descriptor parameter values

Descriptor	Support radius	Descriptor Length	Implementation platform
RoPS	40 $\overline{mr}$	135	MATLAB
SHOT	40 $\overline{mr}$	352	C++ (PCL)
FPFH	20 $\overline{mr}$	33	C++ (PCL)
3DSC	30 $\overline{mr}$	1980	C++ (PCL)
Local D1	40 mr	240	MATLAB
HoD	40 mr	240	MATLAB
Local D1-S	40 mr	40	MATLAB
HoD-S	40 mr	40	MATLAB

Evaluation and comparison of both HoD variants against the descriptors presented in Table 5-1 is performed on three datasets of different point cloud quality. Trials include the high quality Bologna dataset [112], the medium quality SpaceTime [52] dataset and the low quality Kinect dataset [52] (Figure 5- 10). Trials evaluate all competitors for their recognition performance, processing efficiency, compactness and descriptor storage memory demands.



(a)

(b)

(c)

Figure 5- 10 Examples of matching HoD local descriptors in 3D object recognition scenarios. (a) Bologna dataset non-uniformly down-sampled to 1/8 its original resolution with Gaussian noise (σ=30%mr) (b) SpaceTime dataset and (c)Kinect dataset. Green lines show correct matches while red wrong correspondences. Red and blue crosses represent the randomly selected keypoints and their correspondences respectively (b) and (c) are presented with texture information for better viewing.

5.2.4.2 Experimental Setup on the Bologna dataset

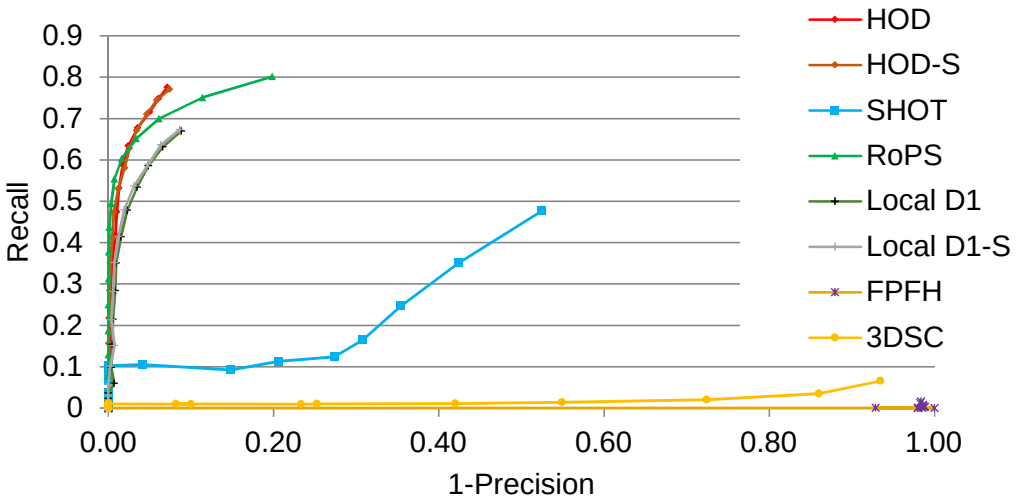
The first set of trials is on the Bologna dataset, which comprises of 6 models and 45 scenes. Models are taken from the Stanford 3D Scanning Repository [198] and are randomly rotated and translated to create clutter and object pose variations. In contrast to [90], this trial exploits the entire Bologna dataset and not only a subset.

5.2.4.3 Robustness to Severe Noise

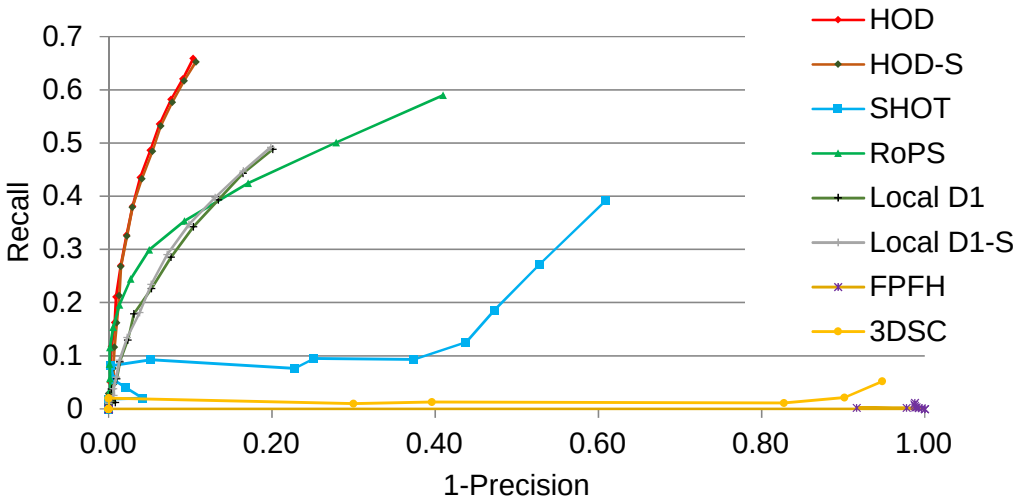
This trial evaluates the robustness of HoD and HoD-S against the descriptors of Table 5-1 under severe Gaussian noise levels with zero mean and $\sigma = \{200\%, 300\%\}\overline{mr}$. Noise levels applied are one of the highest in the literature [60]–[62], [89], [90], [112]. Likewise [90], noise is independently added to each x, y and z-axis for all scene vertices. For each noise level, the PR curve generated is presented in Figure 5- 11.

In both noise trials, HoD, HoD-S and RoPS achieve best performance compared to the rest of the descriptors. For the $\sigma = 200\%\overline{mr}$ case, RoPS achieves a slightly higher recall compared to HoD and HoD-S but both suggested descriptors have the highest precision. In the case of $\sigma = 300\%\overline{mr}$ noise level, HoD and HoD-S outperform all competitors including RoPS. In both noise experiments, HoD and HoD-S have identical performance indicating that for a high-density point cloud both description levels i.e. coarse and fine perform equally well.

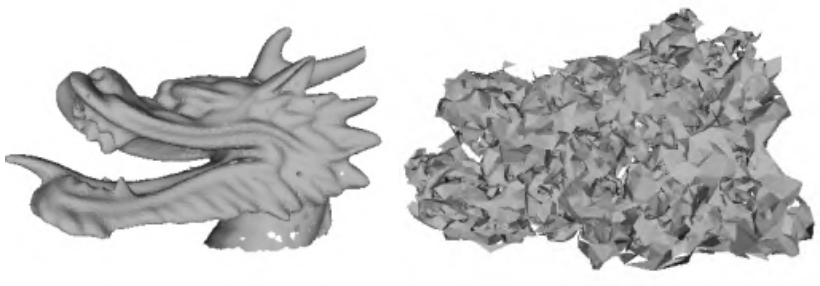
It should be noted that for the top performing descriptors i.e. HoD, HoD-S and RoPS, the computational cost of both HoD variants is much lower compared to RoPS. A detailed processing time analysis is presented in Section 5.2.4.6. Regarding the Local D1 and Local D1-S, it is worth mentioning that they both have a notable performance achieving an acceptable recall and a high precision. SHOT achieves moderate performance while FPFH and 3DSC are very sensitive to such a high noise level confirming the finding in [165].



(a)



(b)



(c) (d)

Figure 5- 11 PR curves under various Gaussian noise levels (a) σ=200% $\overline{mr}$ (c) Original object (b) σ=300% $\overline{mr}$ and (d) object with σ=200% $\overline{mr}$ Gaussian noise (objects are in mesh representation for better viewing)

Both HoD alternatives are robust to noise due to three factors:

c. They neglect the prone to noise LRF/A estimation on which the rest of the descriptors rely.

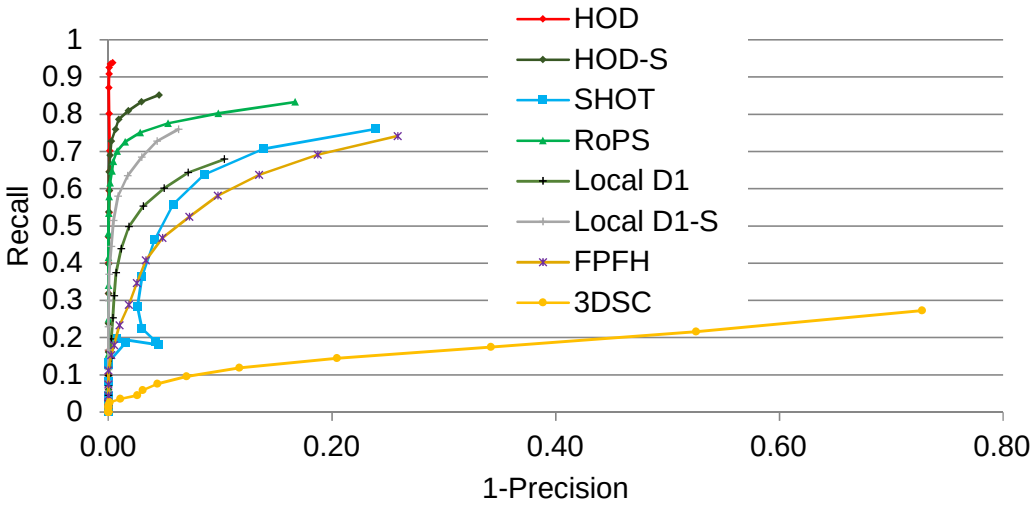
d. By exploiting a border point as a local reference point. Instead of the fixed centroid, as proposed by Osada [105], the suggested reference point offers twice that discriminating capability with an analysis following in Section 5.2.5.

e. By using a sufficiently sized description bin B such as distance fluctuations due to noise still enter the original noise-free bins.

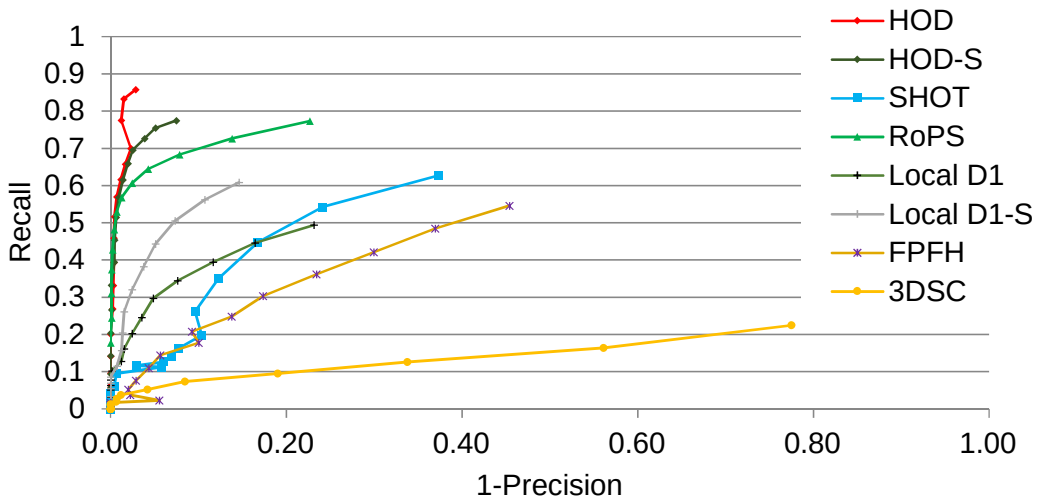
5.2.4.4 Robustness to Non-uniform Subsampling

The robustness of HoD and HoD-S under various non-uniform subsampling levels is challenged against the descriptors of Table 5-1. In contrast to [61], [63], [90], [112] the non-uniform subsampling case is preferred as in reality laser beam distortions can influence the spatial location and the total number of point cloud vertices in an irregular manner. Therefore, the noise-free scenes are non-uniformly subsampled to $\{1/4, 1/8\}$ of their original resolution. For each noise level the PR curve generated is presented in Figure 5- 12.

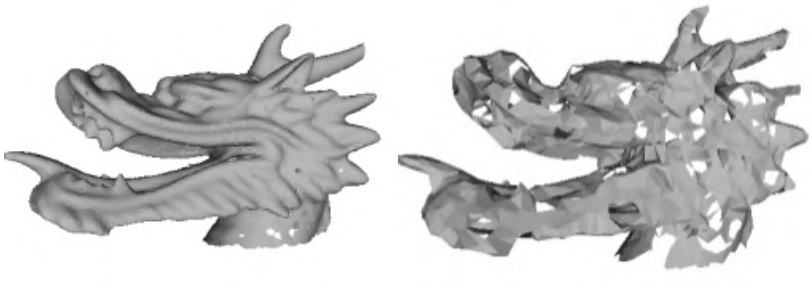
In both subsampling cases, HoD and HoD-S outperform all competitors. In specific, HoD has the highest performance with HoD-S following closely. In contrast to the noise trials, the multi-description level of HoD indeed enhances the recognition performance in the subsampling trials. This is evident as HoD is more robust than HoD-S. Although in the $1/4$ case SHOT and FPFH have a similar performance, in the $1/8$ non-uniform subsampling case SHOT performs slightly better. This is because the LRF of SHOT involves a greater amount of vertices compared to FPFH that takes into account only the k-Nearest Neighbours and their immediate neighbours of each keypoint. Regarding both Local D1 variants, Figure 5- 12 clearly shows that they perform poorer than the suggested HoD and HoD-S descriptors. This performance difference is solely due to the different reference point selection.



(a)



(b)



(c) (d)

Figure 5- 12 PR curves under varying resolution (a) 1/4 (b) 1/8 (c) Original model and (d) 1/8 Non-Uniform Subsampling (objects are in mesh representation for better viewing)

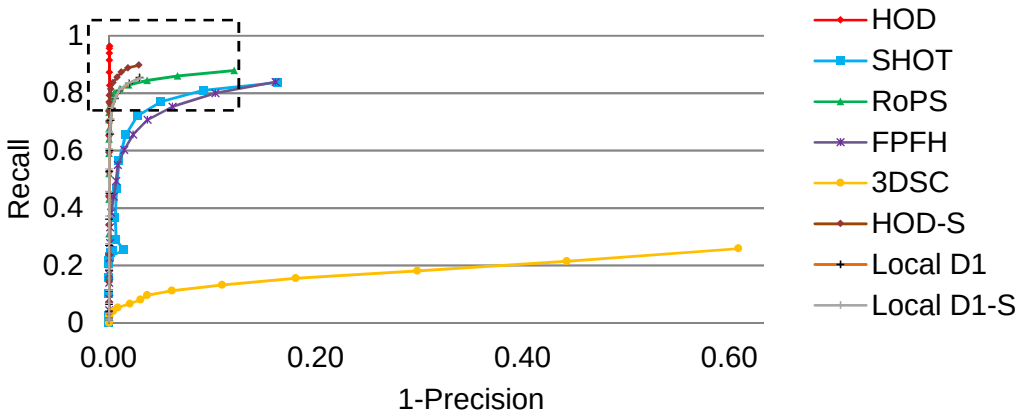
5.2.4.5 Robustness to Combined Gaussian Noise and Non-uniform Subsampling

In the following trials HoD and HoD-S were challenged against the descriptors of Table 6-1 under various non-uniform subsampling and Gaussian noise levels. Trials included $1/2$ subsampling with $\sigma = 10\% \overline{mr}$ and $1/8$ with $\sigma = 30\% \overline{mr}$. For each subsampling – noise level combination the PR curve generated is presented in Figure 5-13.

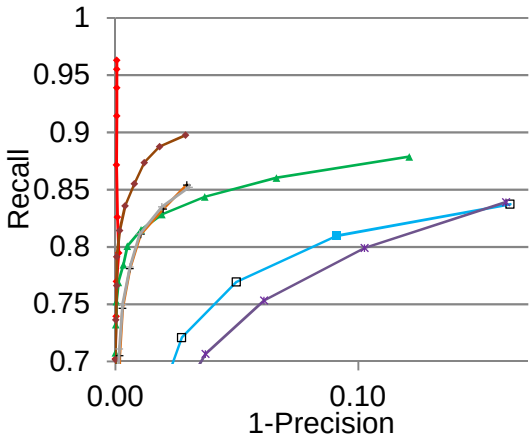
Overall, HoD and HoD-S achieve the highest performance. For the first trial i.e. $1/2$ subsampling with $\sigma = 10\% \overline{mr}$ noise, HoD achieves a remarkable performance. HoD-S follows closely achieving high recall and precision values. The rest of the competitors, excluding 3DSC, although they perform well, they are all inferior compared to both the HoD variants. This is because the LRF they rely on is affected by the high level of combined disturbances applied on each scene. On the contrary, 3DSC presents the lowest recall and precision.

Regarding the $1/8$ subsampling with $\sigma = 30\% \overline{mr}$ noise case, HoD, HoD-S and RoPS achieve equally the highest recall. It should be noted though, that HoD and HoD-S have 15% higher precision compared to RoPS revealing that overall HoD and HoD-S are slightly better. SHOT and FPFH achieve similar recall, with SHOT gaining greater precision. In both trials 3DSC has a poor performance while the Local D1 variants are inferior to both HoD variants, revealing once more the importance of the reference point selection.

It is important to note that in both noise level - subsampling cases, HoD and HoD-S are much more processing efficient compared to the second best performing RoPS, with detailed results presented in Section 5.2.4.6.

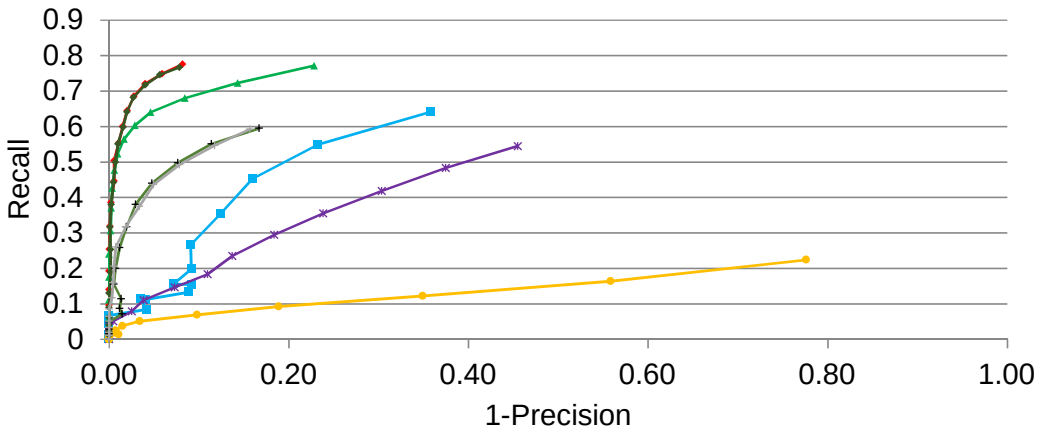


(a)



(b)

(c)



(d)

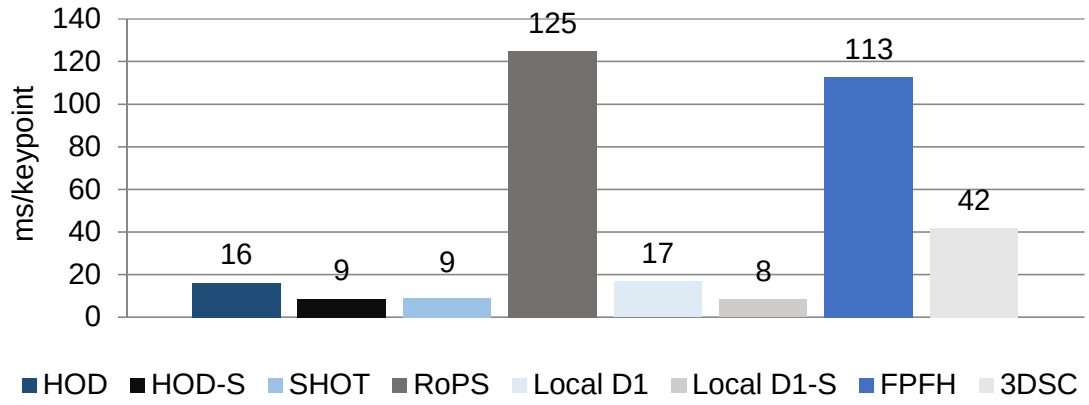
Figure 5- 13 PR curves under combined varying mesh resolution and Gaussian noise (a) $1/2$ & σ=10% $\overline{mr}$ (b) magnified region indicated with a dashed square (d) $1/8$ & σ=30% $\overline{mr}$ (c) objects $1/2$ non-Uniform subsampled with 10% $\overline{mr}$ noise (top) $1/8$ non-Uniform subsampled with 30% $\overline{mr}$ noise (bottom) in mesh representation for better viewing

5.2.4.6 Processing Efficiency

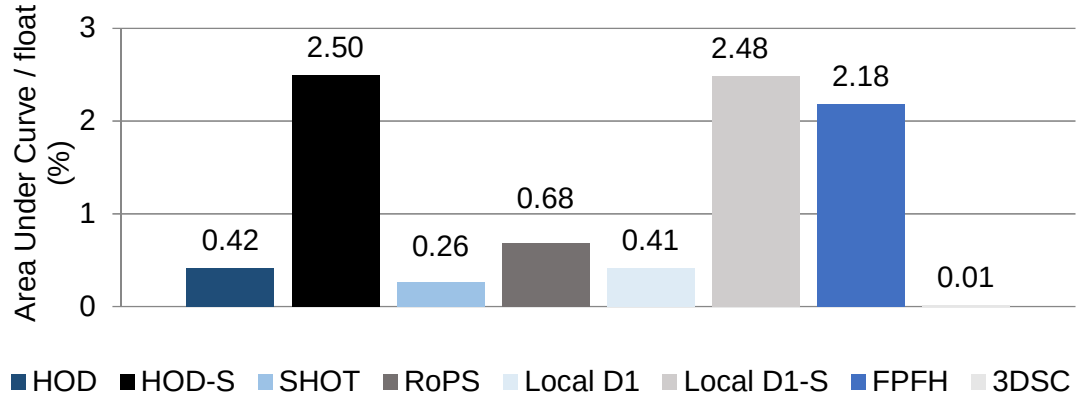
This research focuses on 3D object recognition for time-critical applications. Therefore, it is important to identify the actual processing efficiency of both suggested techniques against the descriptors presented in Table 5-1. Even though HoD and HoD-S include online point cloud resolution estimation and template keypoint description, just by neglecting the LRF calculation they gain a vast processing time improvement. Indeed, HoD-S, Local D1-S and SHOT are the fastest to execute with only 9, 8 and 9ms/keypoint respectively. The reason for the first two is discarding the LRF/A estimation and the small description length while for SHOT it is the C++ implementation. Even their full-sized equivalents i.e., HoD and Local D1, which demand approximately twice that processing time, are still between the fastest solutions purely due to avoiding the LRF/A. As a reminder, HoD, HoD-S, Local D1, Local D1-S and RoPS are MATLAB implemented while the rest are in C++ providing to the former a processing setback purely due to the implementation platform. Despite that, FPFH and 3DSC are considerably less processing efficient even though they are C++ implemented. Detailed results can be found in Figure 5- 14 (a).

5.2.4.7 Compactness

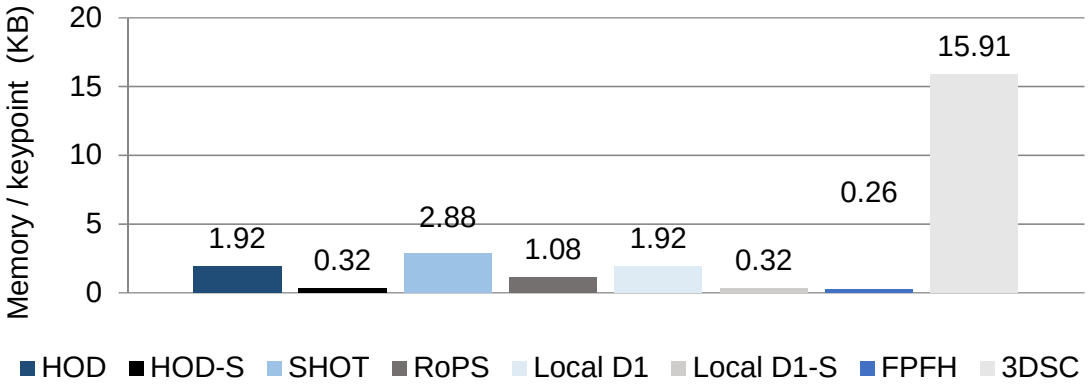
A metric that combines the performance and the descriptiveness of a 3D descriptor is the fraction between the Area Under Curve (AUC) and the number of elements that the descriptor has (float). This metric, named Compactness [57], is based on the AUC from the noise-free PR curve and reveals the descriptive power per element of the 3D descriptor. The higher the value in Figure 5- 14 (b) the more compact the descriptor is. HoD-S and Local D1-S achieve the highest compactness with FPFH closely following. The lowest compactness is presented by 3DSC as it combines low performance with a large float.



(a)



(b)



(c)

Figure 5- 14 Performance evaluation of the HoD / HoD-S with current descriptors (a) Processing efficiency (b) Compactness (c) Storage memory consumption

5.2.4.8 Storage memory Consumption

Another important feature of a missile oriented 3D descriptor is the amount of memory per keypoint it demands for storage. Although it is highly related to the descriptor's length, this trial aims at identifying the specific descriptor storage memory requirement of each descriptor in Kilobytes (KB). FPFH has the lowest memory consumption with the HoD-S and Local D1-S closely following. This is because FPFH is a 33-long descriptor while HoD-S is a 40-element long. Although higher, but still quite low, is the memory demand of HoD. Detailed results for all descriptors under evaluation are presented in Figure 5- 14 (c).

5.2.4.9 Evaluation on the SpaceTime stereo dataset

HoD and HoD-S are further evaluated on the SpaceTime dataset [112] which consists of 6 models and 11 scenes. Compared to the previously tested Bologna dataset, the SpaceTime is more challenging as it includes models and scenes with fewer details. Trials consider the noise-free case with the parameter setup presented in Table 5-1. Texture information is omitted as none of the tested descriptors can exploit it in its current form.

A common conclusion for all competitors is that due to the lower data quality of this dataset, all descriptors perform poorer than previously. Highest recall and precision is achieved by HoD, followed by RoPS, SHOT and HoD-S as shown in Figure 5-15. HoD manages to achieve the highest performance due to its multi-resolution description and matching scheme. Regarding HoD-S, it has a common single resolution description strategy and therefore has inferior performance compared to HoD but is still among the top performing ones. Both variants of HoD perform substantially better compared to the Local D1 descriptors, highlighting the importance of the reference point selection.

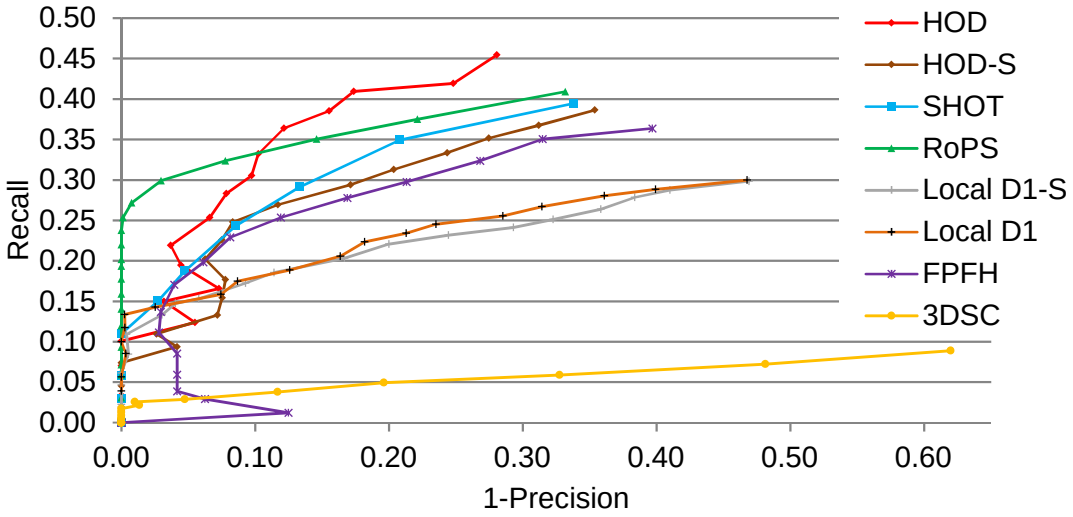


Figure 5- 15 PR curves on the SpaceTime dataset

5.2.4.10 Evaluation on the Kinect dataset

A further evaluation of both proposed 3D local descriptors is on the Kinect dataset [112] which consists of 8 models and 16 scenes. Compared to the previously tested SpaceTime dataset, the Kinect is more challenging as it includes similar models and scenes that have less distinctive details. Equally to the SpaceTime trials, models and scenes are considered texture-less and experiments are performed on the noise-free case with the parameters so far presented in Table 5-1.

Due to the increased difficulty of this dataset, all competitors except HoD present an even lower recognition performance. HoD manages to overcome this challenging situation due to its dual-encoding and dual-feature matching policy. Figure 5-16 clearly shows that HoD has the highest recall and precision by a large margin, followed by SHOT, RoPS and HoD-S.

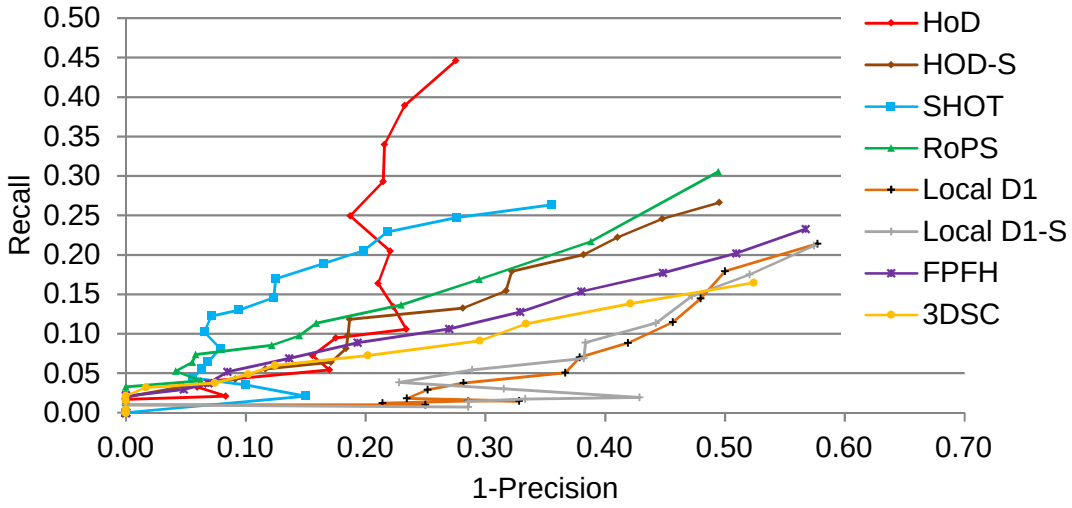


Figure 5- 16 PR curves on the Kinect dataset

5.2.5 Importance of the Local Reference Point selection

From the trials presented in Section 5.2.4, it is evident that the selection of the local reference point is of great importance. Indeed, even though the only difference between HoD and Local-D1 (along with their HoD-S and Local D1-S variants) is the selection of the reference point, their recognition performance varies greatly as shown in Figure 5-17.

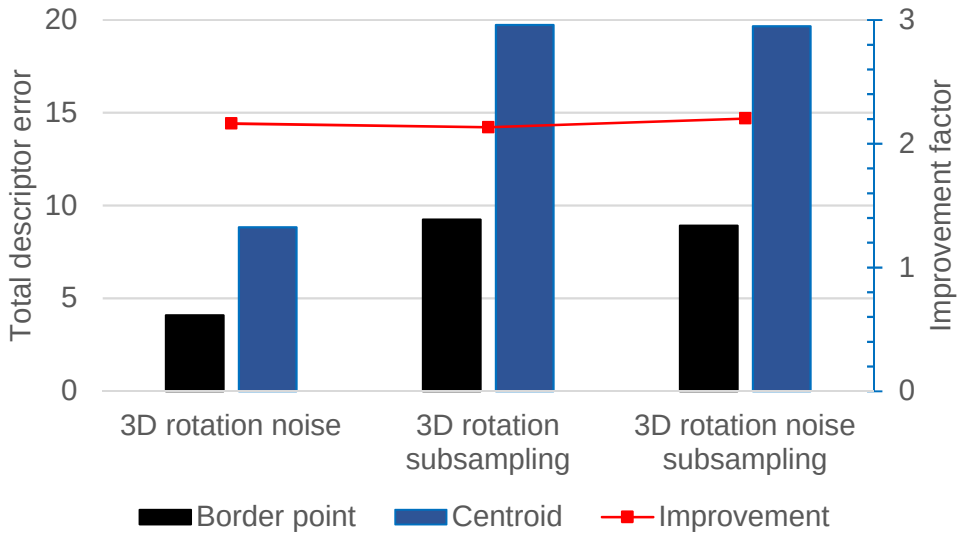


Figure 5- 17 Descriptor error vs. reference point selection on the Bologna dataset

Even though it seems more meaningful to establish the reference point at the supporting area's centroid, as in the Local-D1 case, the description capability of that choice is not optimum. This is because points that belong within a spherical range region are counted in the same bin regardless of their relative position to the reference point. On the contrary, the proposed border point reference point selection has twice that discriminating capability because it encompasses some directional positioning information.

Figure 5-18 shows the case of the centroid and the border based point cloud encoding in blue and in red respectively. For the centroid case (in blue) vertices that fall within the range region rings are counted for the same description bin regardless of their relative position to the reference point i.e. being above or below the centroid. In the border point case (in red), the reference point offers twice the discrimination capability of the former centroid case as by default directional positioning information is taken into account i.e. all vertices are above the reference point.

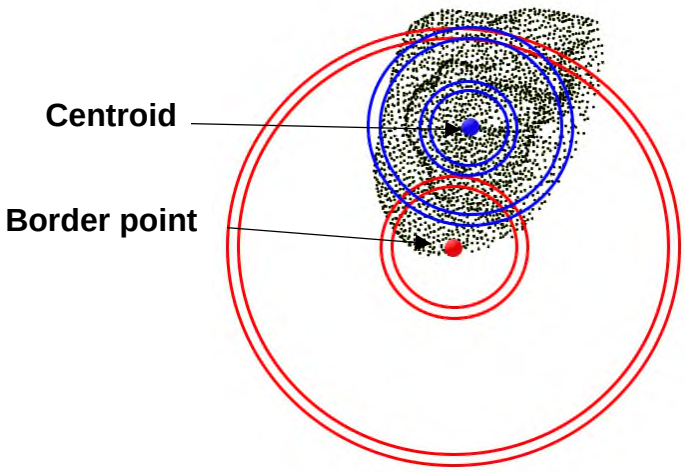
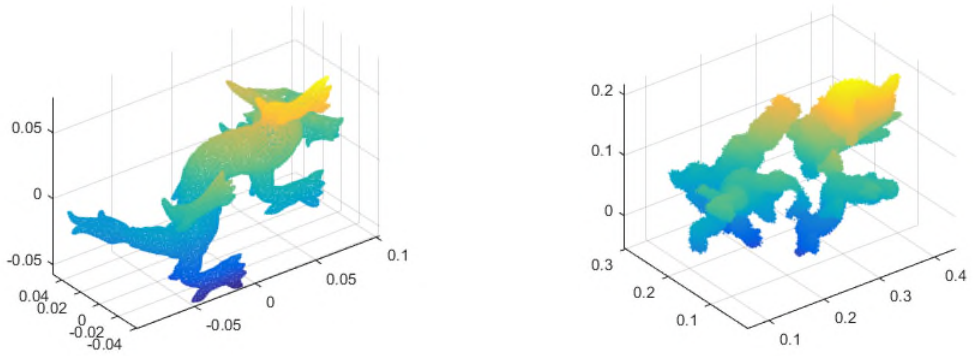


Figure 5- 18 Point cloud encoding based on the reference point selection

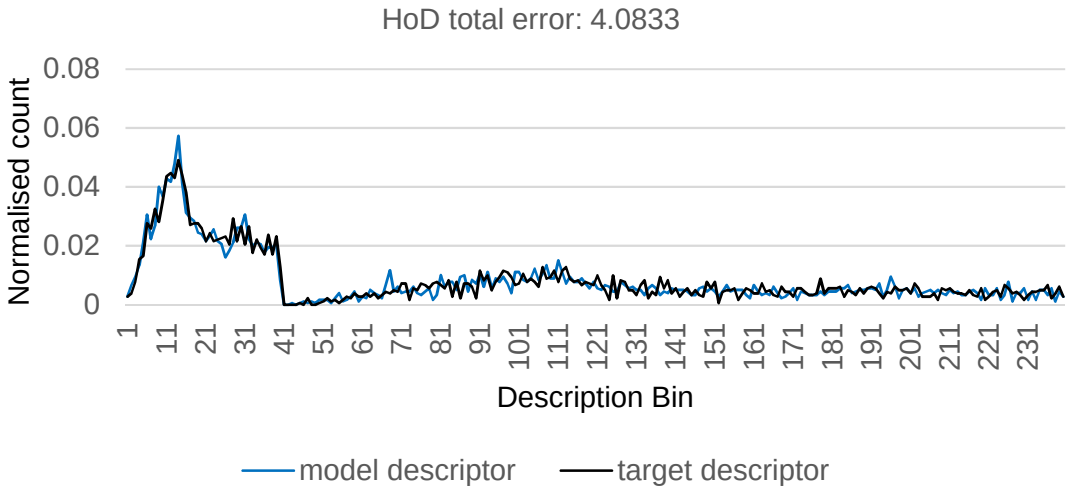
Figure 5-19 to Figure 5-21 challenge the invariance capability of the two reference point options under various perturbations. The cumulative description error for all features shown as the plot title clearly shows that the suggested technique outperforms the centroid based option. In fact, plots (c) and (d) of each Figure show the maximum description error between the template and the scene per

reference point selection, highlighting the accuracy of the proposed strategy under various nuisances.

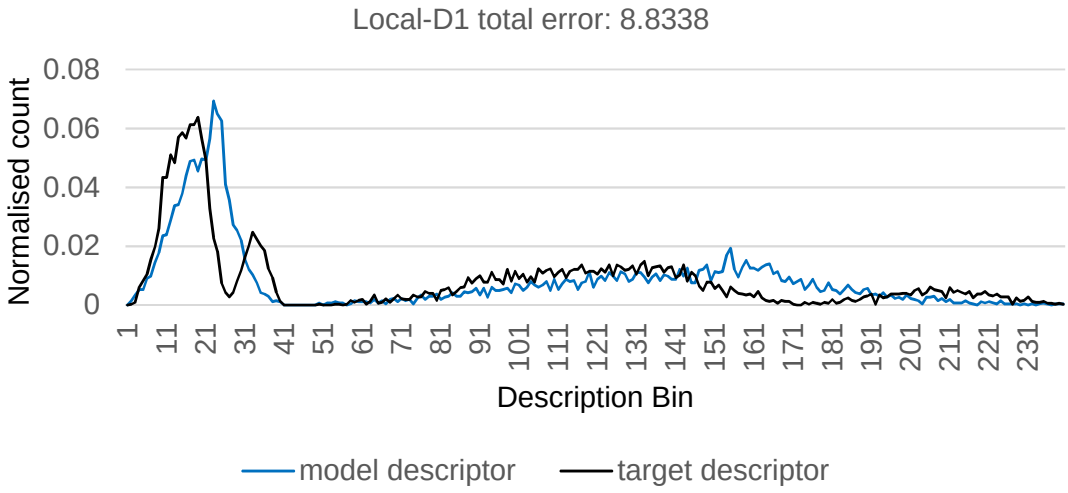
A downside of this reference point selection is assuming the same point during the encryption of each template and each scene support region. During trials on the computer vision datasets i.e. Bologna, SpaceTime and Kinect, the ground truth transformation was already known and thus the same reference point in both the template and the scene could be selected. It should be noted that this methodology is already used in [59], [62], [90], [165]. Obviously, this is not the case for real scenarios and therefore this is compensated by encoding a support region exploiting all possible border vertices as reference frames. This processing bottleneck is compensated by subsampling the support region to maintain the entire processing burden under control. Implementation of this real-world condition is presented in the military scenarios of Chapter 6.



(a) (b)

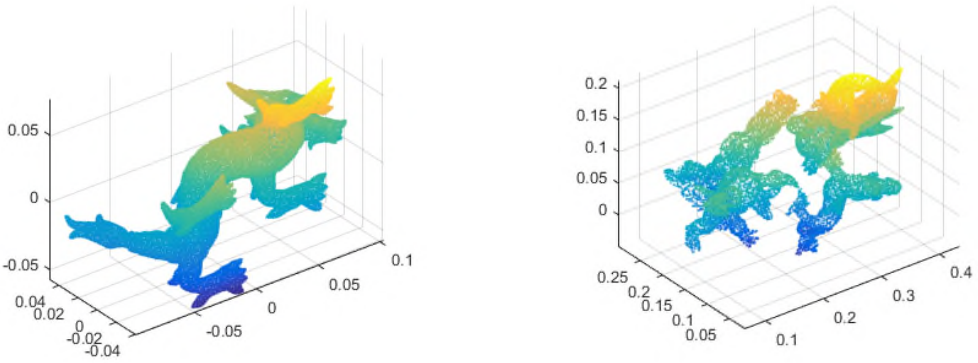


(c)

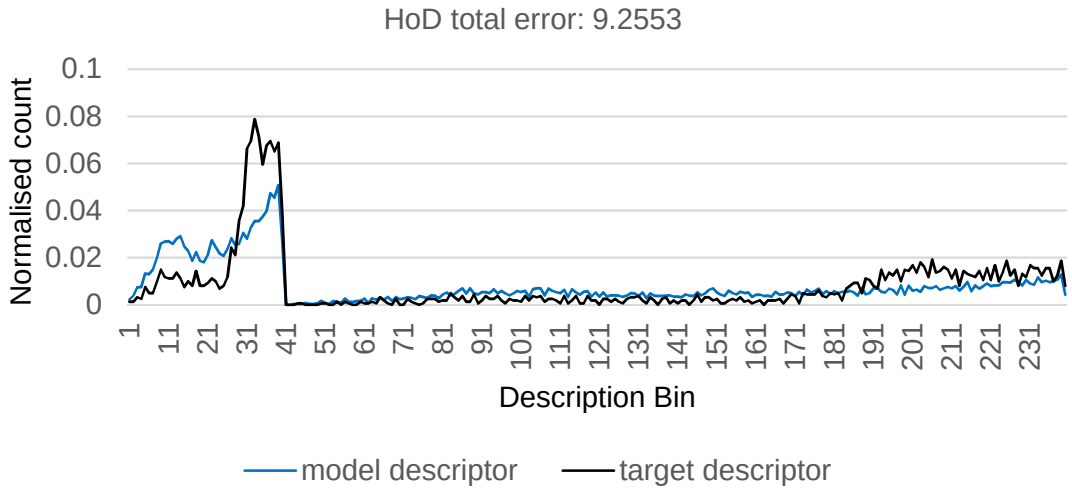


(d)

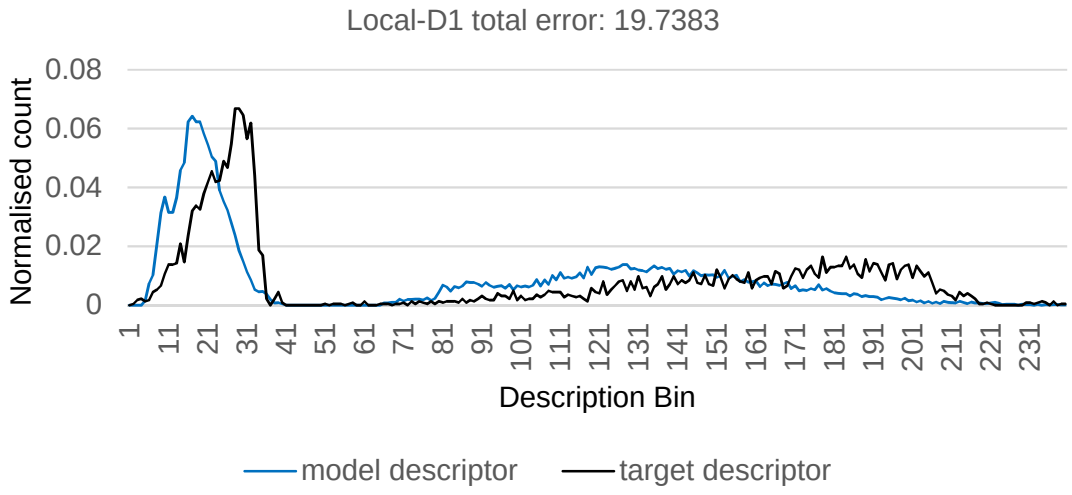
Figure 5- 19 3D rotation and 200%$\overline{mr}$ noise trial (a) model and (b) scene example (c) HoD description error (d) Local D1 description error (Plots show the model – target descriptor correspondence with the maximum error)



(a) (b)

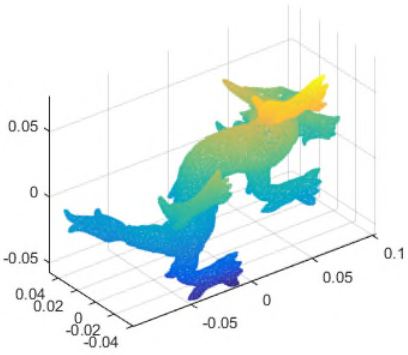


(c)

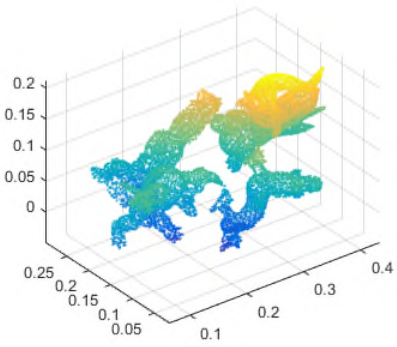


(d)

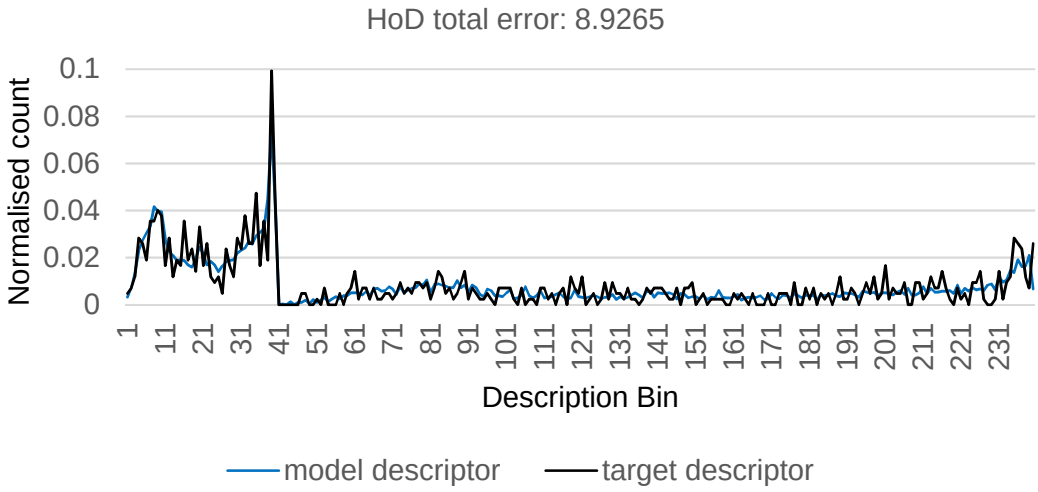
Figure 5- 20 3D rotation and 1/8 non-uniform subsampling trial (a) model and (b) scene example (c) HoD description error (d) Local D1 description error (Plots show the model – target descriptor correspondence with the maximum error)



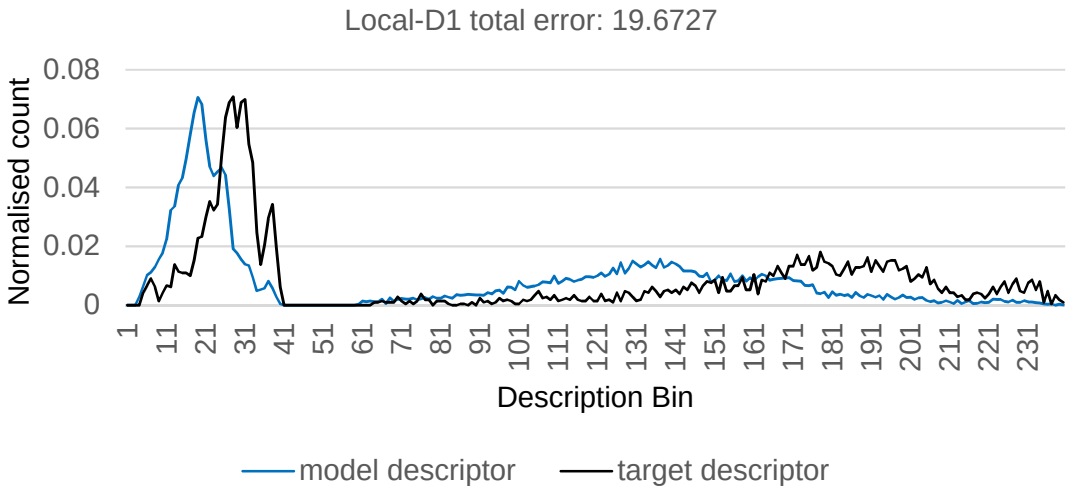
(a)



(b)



(c)



(d)

Figure 5- 21 3D rotation , 1/8 non-uniform subsampling and 30% $\overline{mr}$ trial (a) model and (b) scene example (c) HoD description error (d) Local D1 description error (Plots show the model – target descriptor correspondence with the maximum error)

5.3 Binary HoD (B-HoD)

Although current 3D descriptors are a few (Table 2- 1) and generally perform well, further reducing the processing time and descriptor storage memory requirements is highly desirable but simultaneously quite challenging. Both these requirements can be fulfilled by using a binary descriptor that is created with direct or indirect means. The former involves using either 3D binary descriptors on a point cloud or 2D binary descriptors on a 2.5D range image. Up-to-date 3D binary descriptors are BRAND [100], [101], B-SHOT [143] and [148] which suggests exploiting 2D binary descriptors on a 2.5D image. Although BRAND is claimed to have a good recognition performance, is fast to execute and has a small storage memory demand, it has a feature-level description fusion requirement that encompasses depth and texture information (Section 3.1.3). The latter is not always affordable, constraining BRAND from numerous 3D object recognition tasks. Lately, Prakhya *et al.* [143] transformed the floating point SHOT descriptor into a binary form entitled B-SHOT. This transformation is achieved by forcing four consecutive values of the SHOT descriptor into several sum-based tests that define the binary values of the B-SHOT descriptor (Section 5.1.3.3). Krizaj *et al.* [148] convert the 2.5D scene image into its shape index representation and then apply to it several typical 2D binary descriptors (Section 3.1.1.3). Although their concept is interesting, the shape index calculation introduces a processing burden that affects the total computational time.

Driven by the performance of HoD and the appealing properties of a binary descriptor, the Binary – Histogram of Distances (B-HoD) is proposed. The contributions of B-HoD can be summarized as:

a. A binary descriptor that reduces the processing time and storage memory demand, appropriate for time-critical 3D pattern recognition and registration applications.

b. It combines the processing efficient Hamming distance metric with the well performing Nearest Neighbour Distance Ratio (NNDR) matching criterion. This is unique as current binary descriptors both in the 2D and

the 3D domain exploit the Hamming distance metric in combination with the inferior Nearest Neighbour Distance (NND) criterion.

5.3.1 Establishing the B-HoD descriptor

To reduce the total processing time and the memory footprint of the descriptor, each local area is resampled down to 1/10 its original resolution. Thereafter the HoD descriptor is applied which is then remapped into the binary domain via a Binary-Coded Decimal (BCD) scheme $HoD_{10} \xrightarrow{BCD} B-HoD_2$. Subscripts denote the numerical system each descriptor is based on, which for better readability will be omitted for the rest of this chapter. BCD relies on the gradient of the HOD with the derivative calculated pairwise between the adjacent feature elements. For speedup, derivatives are approximated, hence:

$$B-HoD_k = \nabla HoD = \frac{\partial HoD}{\partial k} \approx \frac{\Delta HoD_{k-1,k}}{\delta} \quad \textbf{(5- 26)}$$

$$\delta = \frac{\max\left(HoD_k\right)}{B} \quad \textbf{(5- 27)}$$

with $k \in \mathbb{N}, 2 \le k \le B$. Finally, the B-HoD descriptor encodes whether the tendency of the gradient is positive or negative, as given by the pseudo code presented in Algorithm 5- 1.

Algorithm 5- 1 Binary Quantization Pseudo Code

```
1   function Binary Transformation
    Input: Floating point number descriptor
    Output: B-HoD descriptor
2   For i=1: HoD descriptor length
3   |   If ∇HoD > 0
4   |   |   B − HoD_k = {1}
5   |   else
    |   |   B − HoD_k = {0}
    |   end
    end
```

It should be noted that creating binary descriptors in an indirect way i.e. remapping a floating-point descriptor via a *BCD* scheme, induces information loss. Nevertheless, it is a generic means to exploit the memory and matching speedup benefits of a binary descriptor.

B-HoD shares with HoD the same features i.e. in contrast to the majority of 3D local descriptors [52], [55], [57], [59], [63], [64], [85], [117], [126], [129], [132], [136] it does not require a LRF/A and it has a dynamically changing support radius contradictory to the majority of current 3D descriptors [60], [63], [64], [81], [100], [112], [117].

5.3.2 Experimental Results

5.3.2.1 Experimental Setup

Given a set of model features f_i^M, a ground truth transformation and the corresponding scene features f_j^S, a scene feature is matched with all model features based on a distance metric and NNDR criterion. If the ratio of the nearest model feature f_i^M with the second nearest $f_{i'}^M$ is less than a threshold τ, then the scene feature f_j^S and the model feature f_i^M are considered as a match.

Based on the established matches, the performance of each descriptor is evaluated in a qualitative manner. Evaluation relies on the estimated transformation matrix T_M based on the model – scene matched keypoints and the ground truth transformation T_{GT}. T_M is calculated based on the ICP algorithm and a point-to-point error minimization metric [202]:

$$T_M = \arg\min_T \left(\sum_{k=1}^{K} \left\| Rm_k + t - s_k \right\|_2 \right) \quad \textbf{(5- 28)}$$

where R,t are the estimated rotation and translation matrices, K is the number of keypoint matches and m_k and s_k are the matched model and scene keypoints respectively. The R,t combination that provides the smallest T is considered as the transformation matrix T_M. Then, given the ground truth transformation T_{GT} between the model and the scene, the T_{error} qualitative measure is calculated:

$$T_{error} = \sqrt{\sum\sum\left(T_M - T_{GT}\right)^2} \quad \textbf{(5- 29)}$$

It is worth remembering that as HoD, B-HoD exploits a multi-level feature matching scheme on each description level i.e., separately for the coarse and for the fine description. The description level that provides most matches is claimed as the accepted domain in which T_M will rely.

During trials, the distance metric used during the NNDR matching criterion for each competitor descriptor is the one originally proposed by each author. The B-HoD descriptor in specific uses the Hamming distance combined with the NNDR matching criterion. Hamming distance matching is further speeded up by fully implementing the matching phase in Boolean arithmetic followed by a bit-count:

$$D_{\text{Hamming}} = \frac{\sum\left(f_i^M \oplus f_j^S\right)}{\sum\left(f_{i'}^M \oplus f_j^S\right)} \quad \textbf{(5- 30)}$$

It is worth noticing that B-HoD is unique in terms of combining the Hamming distance with the NNDR matching scheme since current binary descriptors combine it with the less efficient Nearest Neighbor Distance metric given by:

$$D_{\text{Hamming}} = \sum\left(f_i^M \oplus f_i^S\right) \quad \textbf{(5- 31)}$$

Since a floating-point descriptor is remapped into a lower level binary form, information loss is induced affecting the number of correspondences achieved during the matching stage. Hence, a registration performance drop is anticipated.

During trials B-HoD is challenged against RoPS [59], SHOT [112], FPFH [117], 3DSC [55], USC [89], HoD [21] and a binary version of HOD exploiting the quantization pipeline of [143] in combination with the subsampling of the currently proposed B-HoD descriptor. For better readability, this variant of HoD is notated as HoD (*) throughout this section. It is important to compare B-HoD against HoD(*) to reveal the effectiveness of the suggested *BCD* remapping as beyond that these two descriptors are identical.

The support radius of each descriptor is independently tuned on training scenes from the Bologna dataset [112]. These scenes are non-uniformly down-sampled to ½ their mesh resolution and Gaussian noise is added with zero mean and $\sigma = 10\%\overline{mr}$ [52], [59].

All trials are performed in MATLAB and in C++. Implementations in C++ are attained from PCL Version 1.7.2 [200] while RoPS from MATLAB File Exchange [201]. Beyond the support radius which is tuned for best recognition performance, the rest of the parameters are fixed either to the ones originally proposed by their authors or to their PCL implementation [90]. The tuned parameter settings for all feature descriptors are presented in Table 5- 2. Compared to the rest of the descriptors, FPFH has the smallest support radius with a peaking performance at $20\%\overline{mr}$ confirming the trend stated in [90].

Similarly to the HoD and HoD-S trials of Section 5.2, 100 keypoints from each model are randomly selected and their corresponding ones in the scene are extracted based on their a priori known ground truth transformation T_{GT}. Random keypoint selection is preferred against exploiting a keypoint detector [87] as errors of the detector can affect the descriptor [59].

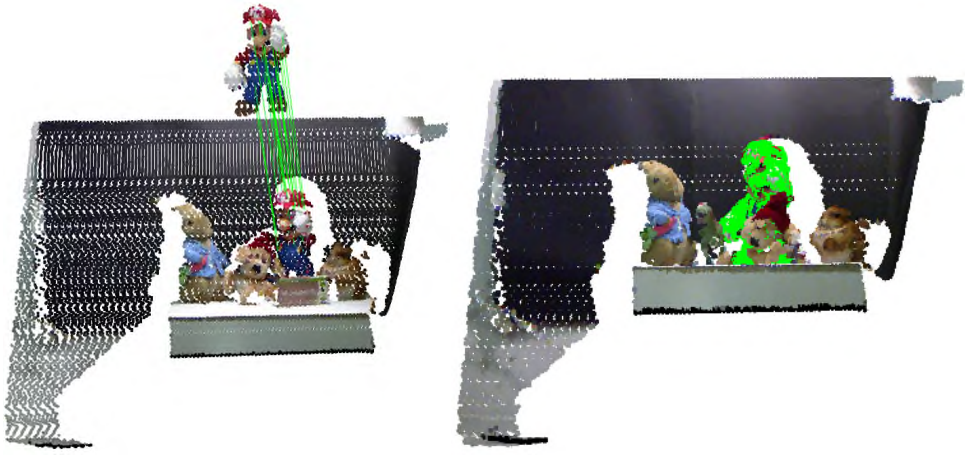
Table 5- 2 Descriptor parameter values

Descriptor	Support radius	Descriptor Length	Implementation platform	Domain
RoPS	40 $\overline{mr}$	135	MATLAB	Floating point
SHOT	40 $\overline{mr}$	352	C++ (PCL)	Floating point
FPFH	20 $\overline{mr}$	33	C++ (PCL)	Floating point
3DSC	30 $\overline{mr}$	1980	C++ (PCL)	Floating point
Local D1	40 mr	240	MATLAB	Floating point
HoD	40 mr	240	MATLAB	Floating point
B-HoD	40 mr	240	MATLAB	Binary
HoD (*) adopting [143]	40 mr	240	MATLAB	Binary

5.3.2.2 Evaluation on the Kinect dataset

Trials are based on the Kinect dataset [112], which comprises of 51 model – scene combinations. Texture information is neglected and the evaluation is based on the T_{error} metric (Equation 5-29). Figure 5- 23 (a) shows the T_{error} of all descriptors, with each peak representing the registration error between the 3D transformation estimated from the keypoint matches and the ground transformation. In specific, Figure 5- 23 (a) reveals that B-HoD, HoD and RoPS present the smallest registration error. HoD(*) and FPFH are next to follow with several spikes of high T_{error} levels. It is worth noting that B-HoD has a smaller T_{error} compared to the HoD(*) indicating that the proposed *BCD* remapping is more accurate compared to the proposed scheme in [143]. Less accurate are SHOT, 3DSC and USC, which attain the highest registration errors.

Focusing on the high performing ones i.e. B-HoD, HoD and RoPS, Figure 5- 23 (b) indicates that B-HoD has almost the same performance as HoD and achieves constantly a lower T_{error} compared to RoPS. A direct comparison between B-HoD and HoD reveals that the performance loss due to the subsampling and the *BCD* remapping is minor, showing that B-HoD is quite promising. A recognition and registration example of the B-HoD on the Kinect dataset is presented in Figure 5- 22.



(a) (b)

Figure 5- 22 Example of B-HoD on the Kinect dataset (a) Green lines indicate correct matches (b) Model point cloud (in green) is registered on the scene point cloud (image from [22]**)**

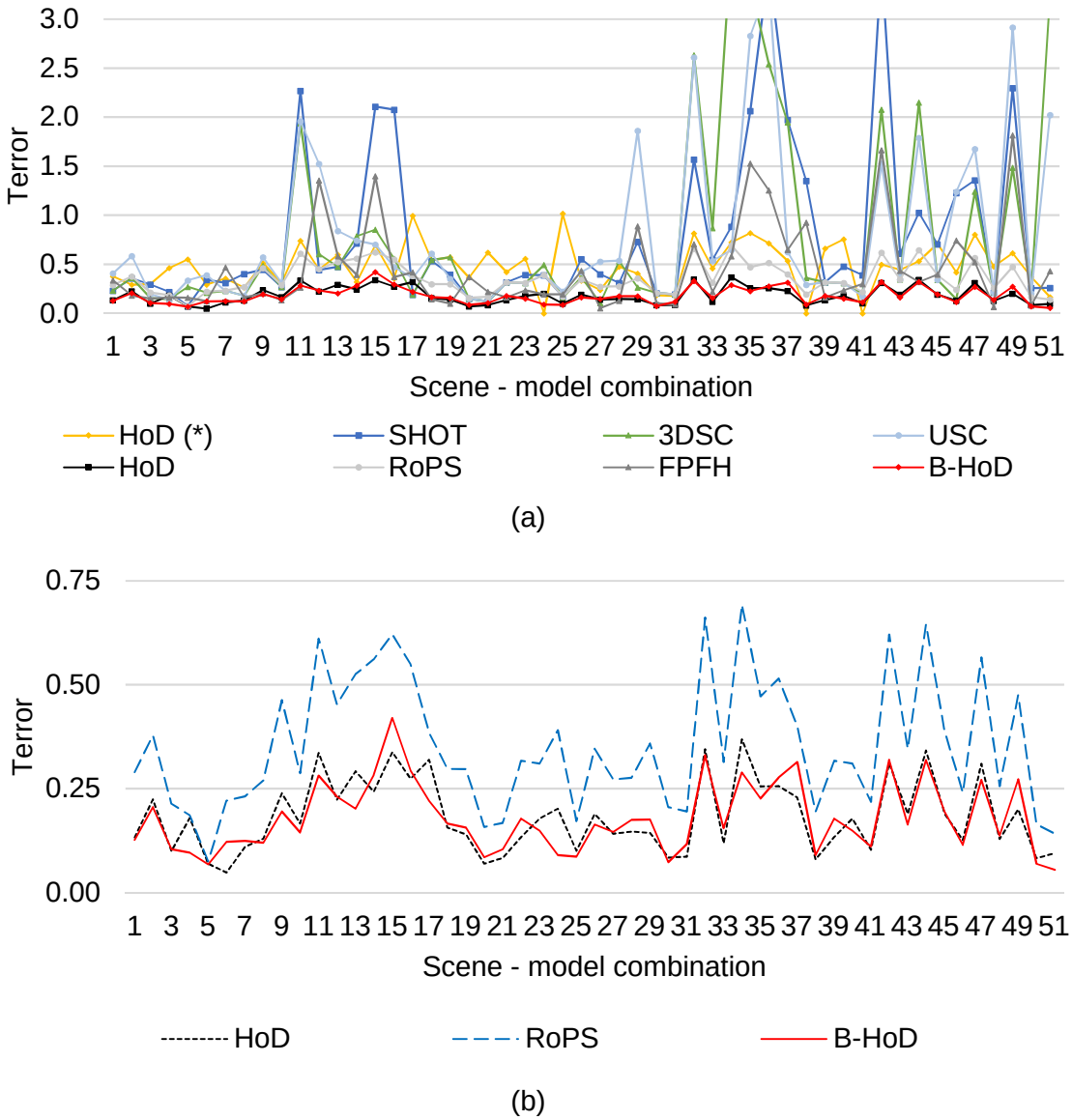


Figure 5- 23 (a) Qualitative performance evaluation based on the T_{error} metric (best seen in colour). Peak values exceeding a T_{error} value of 3 are truncated for better readability (b) Highlighting the top 3 performing descriptors

5.3.2.2.1 Processing Efficiency

This research focuses on 3D object recognition for time-critical applications. Thus, it is important to investigate the processing efficiency of B-HoD against the descriptors presented in Table 5- 2.

Even though all HoD variants include real-time point cloud resolution estimation and template – scene keypoint description, neglecting the LRF estimation reduces greatly the processing time. It is expected though that B-HoD is even further processing efficient due to two additional factors. First, the local area is subsampled and second feature matching is based on the efficient Hamming distance. Indeed, Figure 5- 24 shows that B-HoD is the most efficient 3D descriptor among the ones evaluated with a large margin. A direct B-HoD – HoD comparison reveals that B-HoD is more than 7.5 times faster compared to HoD with a processing time of 0.85ms/keypoint. It is worth noting that all HoD variants and the RoPS algorithm are MATLAB implemented while the rest are in C++ providing to the former a processing setback purely due to the implementation platform. Even in that case, B-HoD is more than x40 faster compared to SHOT which is the fastest one implemented in C++.

For completeness, Figure 5- 25 further analyzes the execution time of each sub-process of the B-HoD and HoD descriptor. The vast processing speedup is obtained via the local area subsampling that is incorporated within the B-HoD. In addition, the NNDR Hamming based matching scheme reduces matching time down to 25% compared to the original floating point NNDR matching scheme.

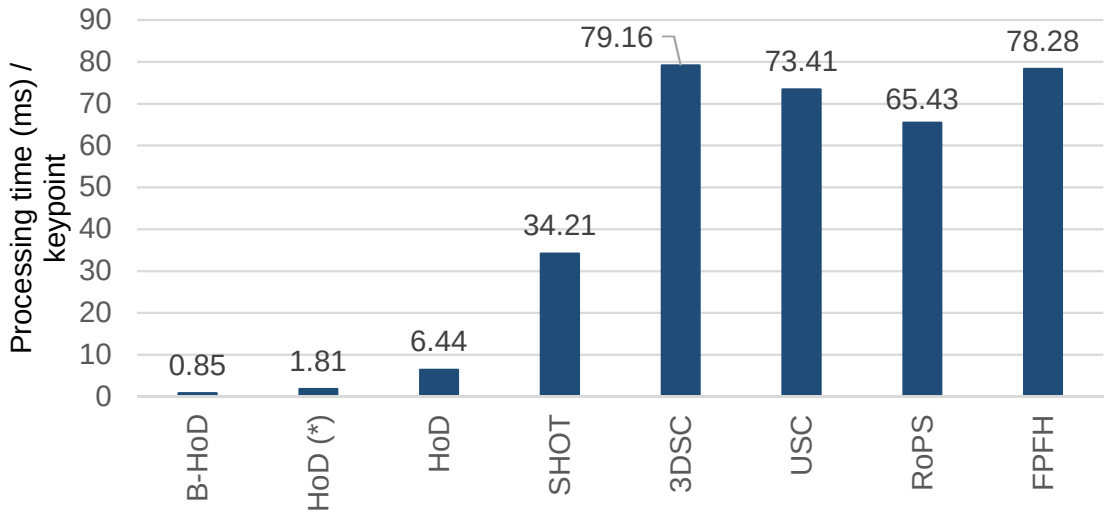


Figure 5- 24 Processing efficiency of the proposed and current descriptors

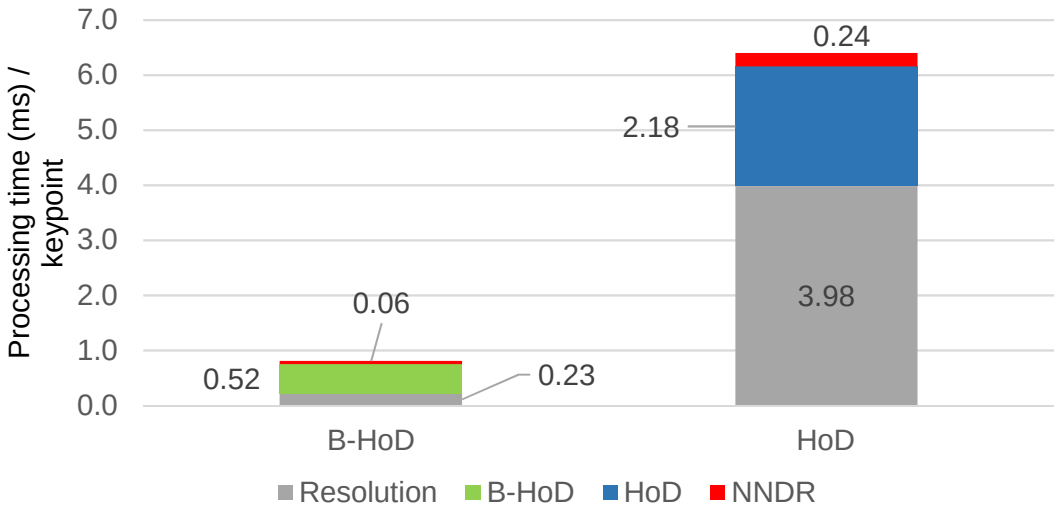


Figure 5- 25 B-HoD and HoD processing time comparison

5.3.2.2.2 Storage Memory Consumption

Another important factor is the memory per keypoint required to store the descriptor. Although memory demand is highly related to the descriptor's size and domain, the specific memory demand in Kilobytes (Kb) per descriptor is shown in Figure 5- 26.

As expected, B-HoD and HoD (*) have the smallest memory footprints of only 0.24 Kb/keypoint due to their binary nature. Although not binary, but purely due to the small descriptor size, FPFH closely follows with 0.26Kb/keypoint. As expected, the binary B-HoD has a reduced memory requirement compared to floating point HoD by a factor of eight. Hence, B-HoD can be considered as highly appealing especially for hardware-constrained platforms.

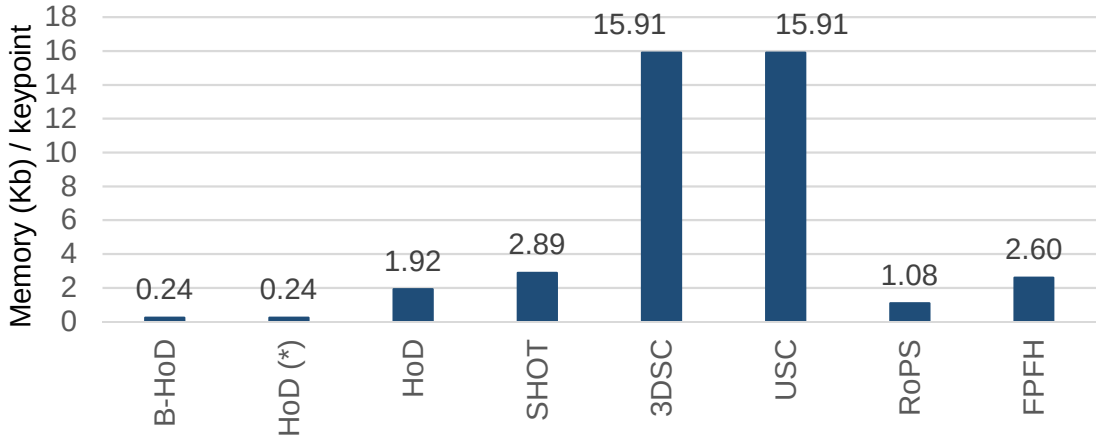
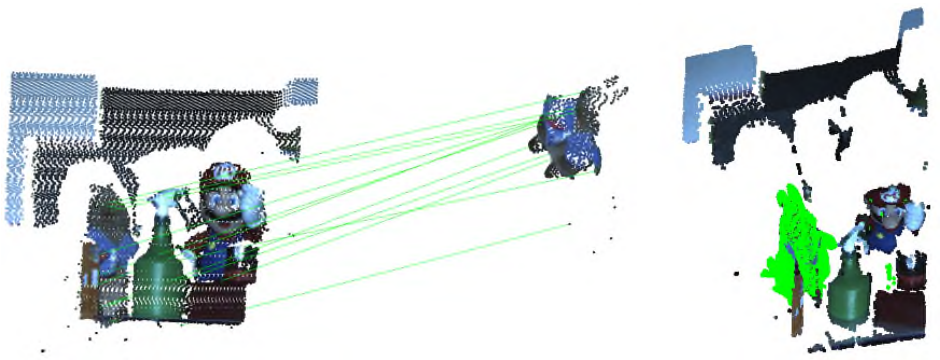


Figure 5- 26 Storage memory requirement per descriptor

5.3.2.3 Evaluation on the SpaceTime stereo dataset

The B-HoD descriptor is further evaluated on the SpaceTime dataset [112] which consists of 24 scene – model combinations. Trials consider the parameter setup presented in Table 5- 2 and texture information is neglected. Figure 5- 27 shows an object recognition and registration example which clearly shows that B-HoD affords an appealing keypoint matching capability that has a low registration error.



(a) (b)

Figure 5- 27 Example of B-HoD on the SpacreTime stereo dataset (a) Green lines indicate correct matches (b) Model point cloud (in green) is registered on the scene (image from [22]**)**

Figure 5- 28 (a) presents the T_{error} per descriptor per scene – model combination. A first conclusion is that all descriptors have an inferior T_{error} metric compared to their corresponding performance on the Kinect dataset, due to the low-quality data of the SpaceTime dataset. B-HoD, RoPS and USC are the ones performing best as they offer the smallest registration error, with the latter having a few T_{error} spikes. Next to follow are HoD, SHOT and FPFH, while less accurate are HoD(*) and 3DSC. Focusing on the high performing ones i.e. B-HOD, USC and RoPS, (Figure 5- 28 (b)) B-HoD has the smallest T_{error} with some minor fluctuations. Indeed, B-HoD achieves the lowest Terror on almost every scene. This is important because the next two best performing ones have a very large processing burden and storage memory requirement compared to the proposed B-HoD, which makes B-HoD a promising solution. A direct comparison between B-HoD and HoD reveals that B-HoD performs better in the SpaceTime dataset. This is because SpaceTime has low quality data and therefore quantizing the histogram of distances into a binary form can compensate for smaller T_{error} values.

A direct performance comparison between the Kinect and the SpaceTime stereo datasets reveals that the performance hierarchy remains almost the same. Another common feature is that B-HoD and RoPS afford constantly a small overall T_{error}. It should be noted though that B-HoD is more than 75 times faster and its storage memory footprint is 4.5 times smaller compared to RoPS.

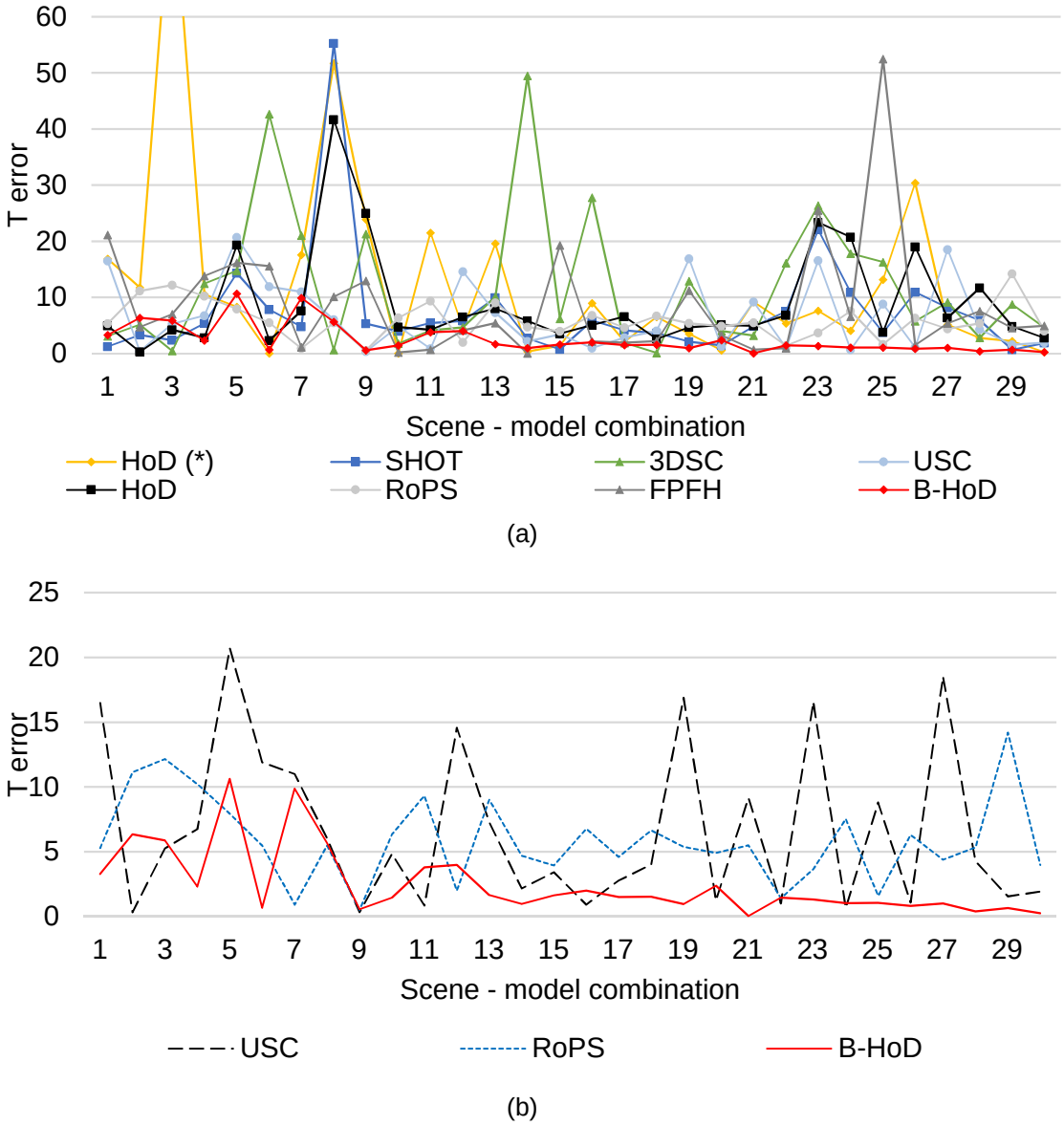


Figure 5- 28 (a) Qualitative performance evaluation based on the T_{error} metric (best seen in colour). Peak values exceeding a T_{error} value of 3 are truncated for better readability (b) Highlighting the top 3 performing descriptors

5.3.3 Conclusion on the B-HoD descriptor

This section introduces a binary 3D descriptor entitled the Binary Histogram of Distances (B-HoD) which is an extension of the HoD presented in Section 5.2. B-HoD is computationally efficient and requires low storage memory resources due to the local area subsampling and efficient BCD remapping scheme.

B-HoD is challenged with several local 3D descriptors, including state-of-the-art ones, on two popular low-resolution datasets, the Kinect and the SpaceTime stereo. B-HoD maintains a registration error to a lower level via an efficient BCD remapping scheme that exploits the NNDR match metric in combination with the Hamming distance. Specifically, B-HoD is one order of magnitude faster and requires one order of magnitude less storage memory compared to the already fast floating-point HoD descriptor.

Based on the low registration error, appealing speedup achieved and on the small storage memory requirements, B-HoD can be considered as an attractive solution for time-critical applications.

5.4 Conclusion

In this Chapter three novel 3D local based descriptors are suggested. The first one considers a computationally efficient local based 3D descriptor, named the Histogram of Distances (HoD) that is based on the local point-pair distance distributions. Main features are the LRF/A independence combined with a multi-encoding and multi-feature matching policy. Both attributes enhance the robustness of HoD to perturbations like noise and subsampling even when the target is under clutter and occlusion. In addition, by neglecting the necessity of a LRF/A estimation, a substantial processing time speedup is gained.

Specifically, this chapter introduces two variants of HoD with different description sizes depending on whether recognition performance or further processing efficiency is aimed and reliant on the quality of the point cloud dataset. Hence, beyond the full HoD descriptor, a coarse variant named HoD-S is proposed. In any case, both alternatives offer a notable high performance in an appealing small execution time suggesting a promising solution for time-critical 3D object recognition applications.

The third 3D local descriptor suggested is the binary variant of the HoD, named the B-HoD. This descriptor has an even smaller execution time while the storage memory it requires is even lesser due to its binary nature. Trials on medium/ low quality datasets revealed its promising performance.

HoD can be considered as an appealing local based 3D descriptor for five reasons:

a. It has a simplistic architecture and thus is appropriate for time-critical applications.

b. HoD neglects the requirement of a LRF/A and thus the computational time is vastly reduced.

c. The support area around each keypoint to be encoded is dynamically changing based on the needs of each scene.

d. Shifting the reference point selection to the border enhances descriptiveness.

e. Robustness to noise and/or subsampling is achieved via exploiting a X^2 match metric in combination with a multi-encoding and multi-feature matching policy.

6 Trials on Military Scenarios

THREE-dimensional ATR implemented on future Light Detection and Ranging seeker missiles, can substantially improve the missile's effectiveness against camouflage, concealment, and deception techniques. Hence, this chapter introduces a standard 3D object recognition pipeline that is accordingly extended to meet the requirements of missile oriented 3D ATR scenarios. This architecture is then used as a testbed to evaluate the current and the suggested 3D descriptors on simulated but highly credible air-to-ground missile engagement scenarios with the missile being under various obliquities, distances to the target and sensor perturbations. Additionally, a single-template concept is implemented for evaluation of all the descriptors.

6.1 Background

ATR for military applications has been extensively investigated for decades. The quest for such automated procedures arises from the demand to reduce the amount of collateral damage and fratricide targeting. Therefore, future LIDAR seeker missiles with ATR capabilities must have a high-true and low-false positive recognition rate to avoid incorrect targeting. In addition, the missile data acquiring subsystem (seeker) and the guidance section of a LIDAR based missile need to have reduced computational cost, and a resistance to countermeasures such as smoke or camouflage type obscuration. Furthermore, the image matching system needs to adapt to the change in scale (as the missile closes on the target) as well as the change in orientation (as the missile manoeuvres during target acquisition and tracking phases of the engagement). Moreover, the recognition procedure must be in real-time. Hence, the afforded processing time for a missile to perform

ATR under these demanding conditions is quite strict. These demands take place in a noisy battlefield environment with a great number of non-targets (clutter) such as non-military vehicles, ground, trees and buildings. that the missile must avoid. In terms of hardware, the computing and sensor unit needs to fit into the missile's guidance section, which necessitates a high packing density for the sensor and the processing electronics.

During the past years, ATR has been investigated in numerous spatial and data domains such as 2D IR [1]–[3], [15], 2D SAR [6], [7], [203]–[205] and Inverse SAR (ISAR) [6], [8] and lately in 3D laser based solutions [9]–[13], [72].

Driven by the appealing advantages of 3D ATR analysed in Section 2.2, the suggested local based descriptors HoD and HoD-S introduced in Section 5.2 are challenged against high-performing 3D descriptors from the computer vision domain on a number of challenging complex missile engagement scenarios. Since real military data are classified, trials in this thesis involve simulated but highly credible air-to-ground engagement scenarios. The dataset used is very challenging as it is realistic, cluttered, occluded, incorporates sensor noise, the target scene is generated under various obliquities (target viewing angles), and laser atmospheric disturbances and variable missile-target ranges are simulated. It should be noted that, although the 3D descriptors evaluated here are of high-performance, the complexity of the missile engagement scenarios does not allow simplistic matching procedures i.e. directly matching the template features against the target's one. Therefore, an ATR recognition pipeline that incorporates an extensive pre and post-processing operations is mandatory.

The significance of this chapter is:

a. The military dataset used is more challenging compared to the current open source literature as it combines a great number of missile and scene parameters.

b. Compared to the literature, it exploits military scenarios while current surveys have a computer vision context [90], [141], [165], [206], [207].

c. Trials simulate scale changes and atmospheric disturbances to the LIDAR laser beam. Both these features are unique, as they have not been investigated previously.

6.2 3D Local Feature Descriptors

A great number of local feature based 3D descriptors exist and several were analysed in Section 5.2. Latest approaches are either Histogram or hybrid Signature-Histogram due to their enhanced robustness. According to [112], Histogram based descriptors describe the local area by accumulating topological characteristics such as vertices, while the Signature descriptors encode the local geometric features like coordinates. While signature based descriptors are more descriptive, the Histogram based ones are more robust to perturbations such as noise, because they compress the extracted information into distinctive bins. Thus a hybrid Signature-Histogram offers both the advantages. The descriptors evaluated in this chapter are the 3DSC [55] (Section 5.1.4.1), USC [89] (Section 5.1.4.2), FPFH [117] (Section 5.1.6.2), SHOT [52], [112] (Section 5.1.3.1) RoPS [59]–[61] (Section 5.1.2.1), and the proposed HoD and HoD-S (Section 5.2), as presented in Figure 6- 1.

6.3 3D ATR Pipeline

The 3D ATR pipeline relies on a typical computer vision 3D object recognition architecture [175] that is properly extended to facilitate the requirements of military scenarios.

Considering that the intended application is a future 3D ATR LIDAR based missile, this Chapter investigates the extreme case of introducing a single template per target instead of multiple partial views. Although the typical multi template view [175] (Figure 6-2) can be implemented, its increasing processing burden is not appreciated. The latter holds true, as during the preliminary trials conducted on a multi-template view scheme, even the fastest HoD-S descriptor required a few seconds per scene to fulfil the ATR pipeline, while SHOT was the slowest, requiring approximately 20 minutes per scene. SHOT required the extended processing time due to the vast number of matches the NNDR criteria

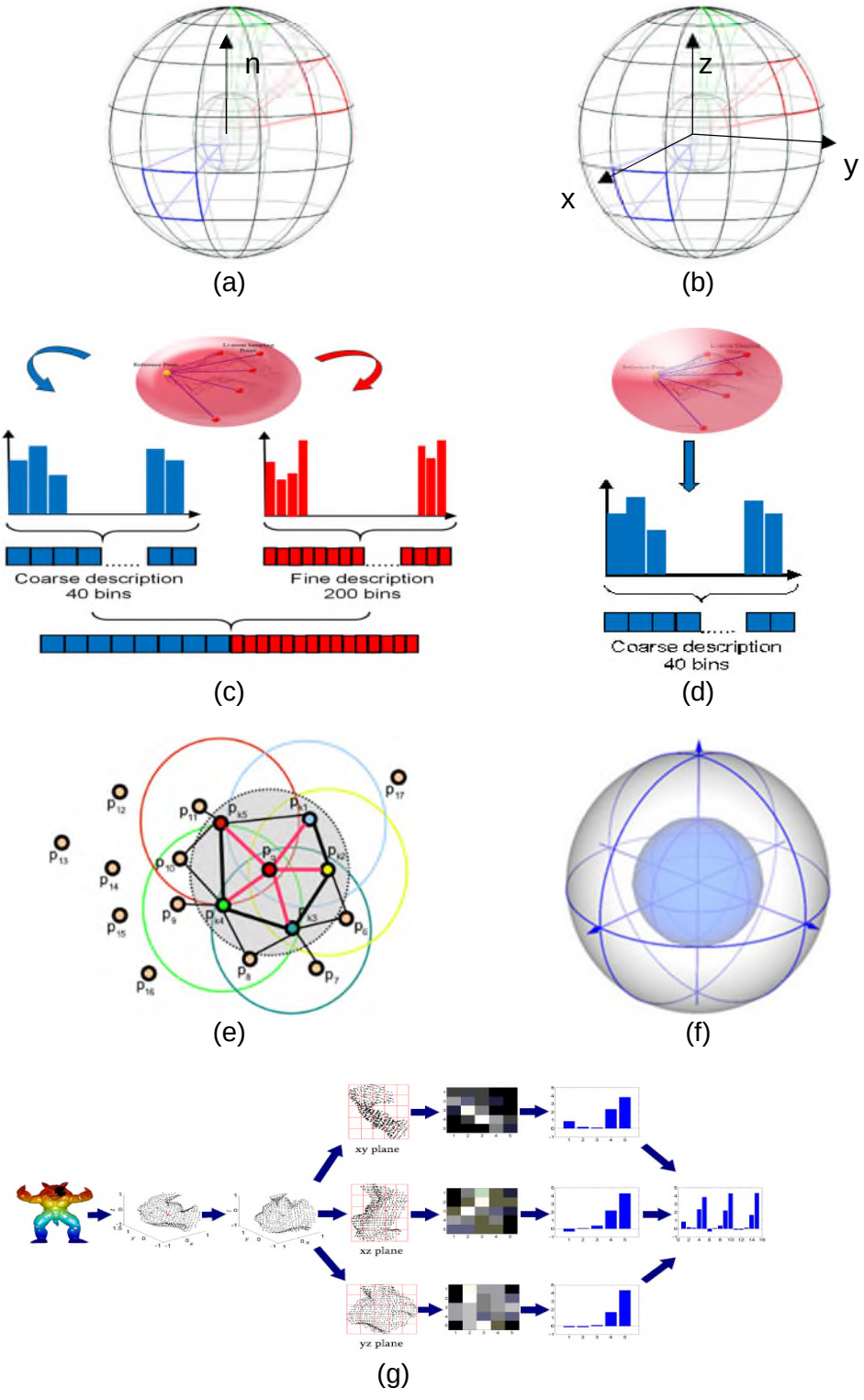


Figure 6- 1 Local feature based descriptors. (a) 3DSC (b) USC (c) HoD (d) HoD-S (e) FPFH (f) SHOT (g) RoPS (all images except (c)-(d) are obtained from the original papers)

produced (in the order of 1000 per scene) that affected the entire ATR process accordingly. On the contrary, the single template scheme is appealing as it is expected to reduce the processing time demand as well as to reduce the storage memory demands.

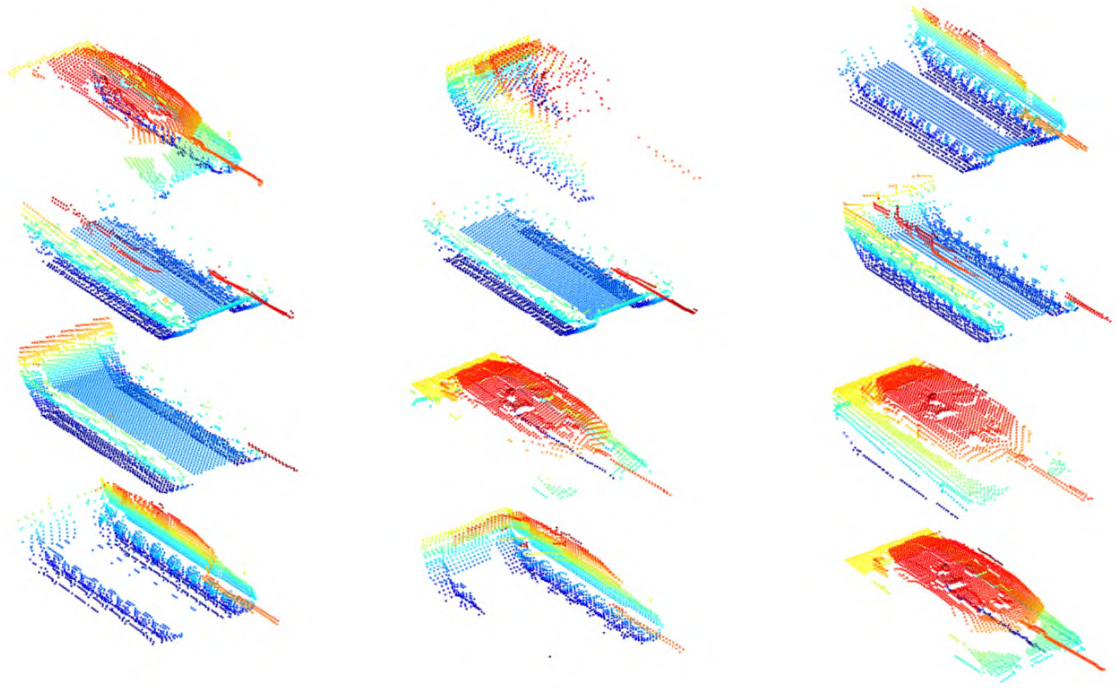
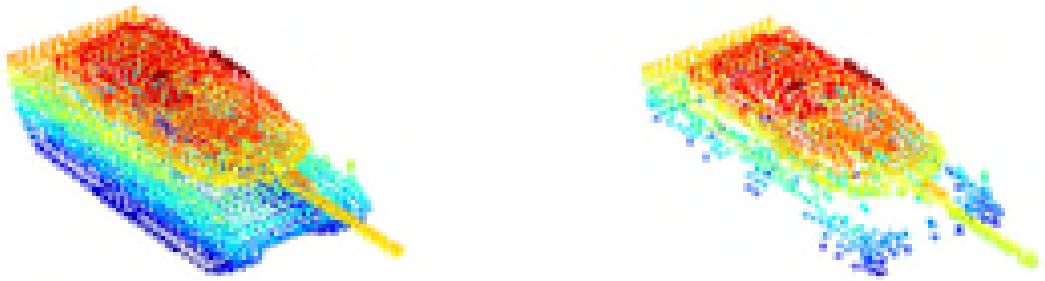


Figure 6- 2 Colour coded partial views of the Leopard C2 MBT (red is closer and blue is further from the virtual LIDAR sensor)

6.3.1 Offline phase

During the offline stage, the input is a 3D point cloud $\boldsymbol{P_m}$ of a Leopard C2 MBT to be recognized. Since a bottom-up viewing orientation of the target is not applicable, the lower part of $\boldsymbol{P_m}$ is rejected by applying the Hidden Point Removal algorithm [157] analysed in Section 4.2.4. The remaining part $\boldsymbol{P_{pv}}$ which is approximately 80% of $\boldsymbol{P_m}$, is shown in Figure 6-3.



(a) (b)

Figure 6- 3 Colour coded MBT (a) ideal point cloud (b) HPR processed (image from [23]**)**

For processing efficiency, the partial point cloud view $\boldsymbol{P_{pv}}$ is then uniformly subsampled at 0.3-meter resolution. Although a keypoint detection strategy can be exploited, for simplicity, all vertices of the subsampled $\boldsymbol{P_{pv}}$ are described with the 3D descriptors of Section 6.2 with a description radius of 15*mr* where *mr* is the average template resolution [59], [63], [112] (for HoD and HoD-S it is the scene resolution [21]).

Considering the large amount of points to be described, all descriptors are assembled into a FLANN [88] structure that will be used during the matching stage. FLANN performs fast approximate nearest neighbour searches in high dimensional spaces by selecting automatically from the libraries it contains, either a randomized kd-tree or a hierarchical k-means tree indexing algorithm, to speed-up the matching process. In both cases a set of the pre-defined parameters is automatically selected that performs best for the evaluated data.

Finally, an ideal 3D point cloud of the MBT is also subsampled at 0.3-meter resolution and is stored for the hypothesis verification stage described in Section 7.3.2.3.

6.3.2 Online phase

The input to the online phase is a scene point cloud $\boldsymbol{P}$ that is uniformly

subsampled at 0.3-meter resolution into $\boldsymbol{P_c}$ that comprises of the vertices $P_c,\{c \mid c \in \mathbb{N}, c < M\}$ where M is the total number of points belonging to $\boldsymbol{P}$.

6.3.2.1 Smooth surface filtering

For each vertex P_c, a normal is associated by estimating the best fitting local plane to its six closest neighbours. Then, for each P_c acting as a centroid, a spherical volume $\boldsymbol{V}$ is extracted with radius equal to the MBT length i.e. 10m. For each $P_d,\{d \mid d \in \mathbb{N}, d < c\}$ belonging to the volume $\boldsymbol{V}$, the standard deviation of the normals enclosed $\sigma(n_{P_d})$ is calculated. Finally, the following cost function defines whether P_c will be part of the filtered point cloud scene $\boldsymbol{P_f}$:

$$P_f = \begin{cases} P_c & \textit{accepted if } \sigma(n_{P_d}) > 10^\circ \\ P_c & \textit{rejected} \text{ elsewhere} \end{cases} \quad \textbf{(6- 1)}$$

It is worth noting that a simplistic cost function accepting vertices by comparing n_{P_c} with the average normal of its surrounding vertices has a questionable performance because it is not robust even to minor perturbations. An example is shown in Figure 6- 4.

Smooth surface filtering discards a number of clutter objects from the scene and thus reduces the overall processing burden and improves recognition performance. Despite that, surprisingly current military oriented literature either does not exploit a noise and smooth surface filtering procedure at all [10] or discards only a planar ground surface [9], [75].

6.3.2.2 Keypoint description, matching and consistency checks

The scene vertices $\boldsymbol{P_f}$ are then described by the descriptors of Section 6.2. For the feature matching, [104] is extended and thus the k-Nearest Neighbour Distance Ratio (kNNDR) with k=10 is used aiming at de-correlating matches from the match metric used by shifting the matching burden to a number of Geometric consistency checks [40], [104].

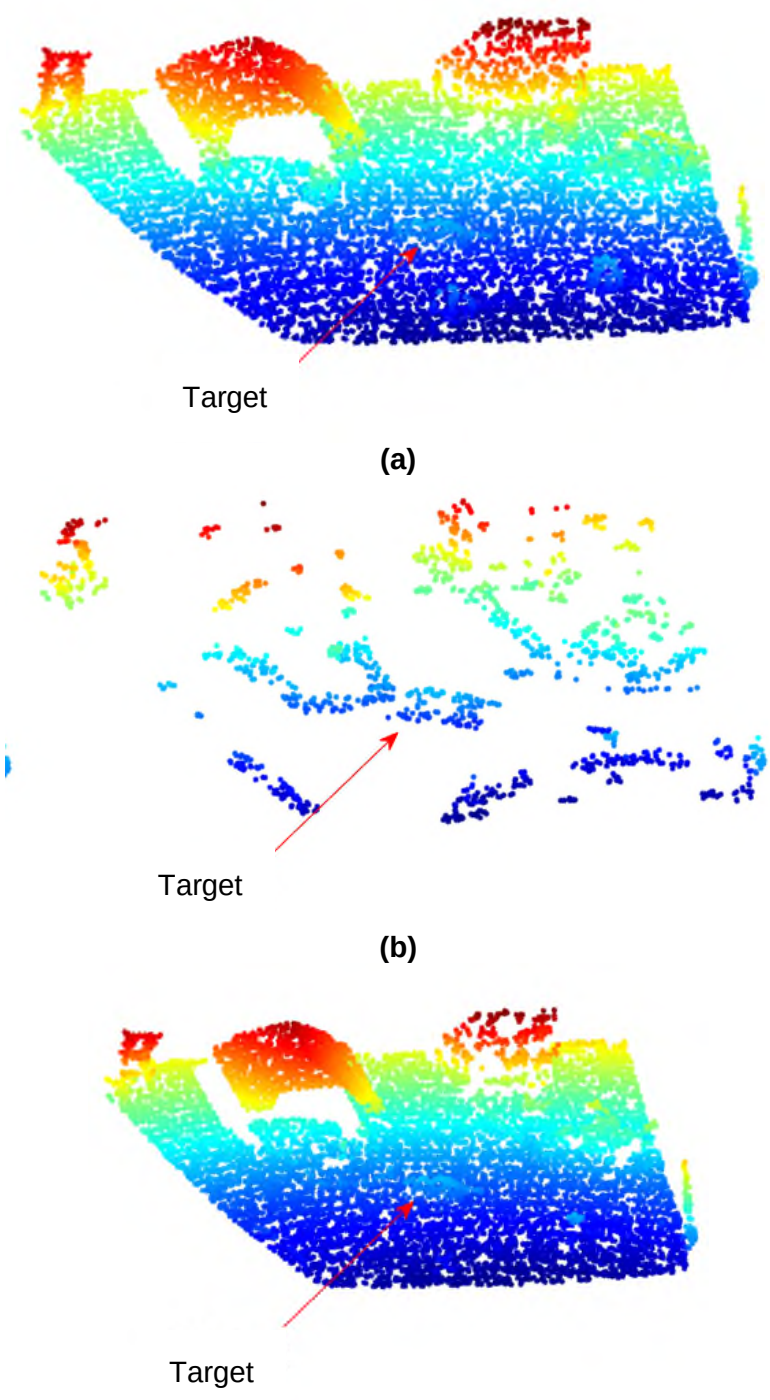


Figure 6- 4 (a) LIDAR point cloud with σ=10cm Gaussian noise (b) proposed standard deviation filtering (c) average smooth surface filtering (height-related colour coding for better visualization) (image from [23]**)**

In specific, during the feature matching stage a template feature f_i^M and a scene feature f_i^S are matched if their k-NNDR metric fulfils the criteria:

$$f_i^M \triangleq f_j^S \longleftarrow\!\!\!\!\!\!- k-NNDR = \frac{\left\| f_i^M - f_j^S \right\|_{metric}}{\left\| f_{i\_idx}^M - f_j^S \right\|_{metric}} < \tau \qquad \textbf{(6- 2)}$$

where *metric* denotes the distance metric proposed by the author of each descriptor, *i,j* are the feature indexes and $f_{i\_idx}^M$ refers to the $idx = \{2,3,..,10\}$ best matching feature. For example, for the $f_{i\_5}^M$ case, the matching criteria evaluates the i^{th} model feature with the j^{th} scene feature, over the fifth best matching model feature model with the j^{th} scene feature. The quality metric that determines the *best matching ranking* is based on the Sum of Square Differences among the j^{th} scene feature and all the model features. To reduce the dependency between the threshold value and the match metric used, the threshold value is set to $\tau = 1$ and the matching burden is shift to a number of Geometric consistency checks.

Geometric consistency checks aim at reducing mismatches by grouping the correspondences into $\boldsymbol{H}$ clusters that are geometrically consistent [40], [104]. Specifically, the k-NNDR matches are grouped into one of the clusters $H_a, \{a \mid a \in \mathbb{N}\}$ with $H_a \in \boldsymbol{H}$:

$$H_\alpha = \left\{ \boldsymbol{P}_{pv}^M, \boldsymbol{P}_f^S \right\} \longleftarrow\!\!\!\!\!\!- \left\{ f^M \triangleq f^S \right\} \qquad \textbf{(6- 3)}$$

where $\boldsymbol{P}_{pv}^M, \boldsymbol{P}_f^S$ are the model and the scene correspondence sets respectively that belong to cluster H_a. The number of clusters α is not pre-defined but can vary depending on the template – scene pair similarity.

The Geometric consistency checks are done as follows. Given a seed correspondence from H_a, the first cluster is initialized and all correspondences $P_{pv\_idx}^M, P_{f\_idx}^S$ with *idx* not yet grouped that are geometrically consistent with the

cluster H_a are added to it. The consistency check for a pair of correspondences $\{P^M_{pv\_idx}, P^S_{f\_idx}\}$ is valid if:

$$\left| \| P^M_{pv\_idx1} - P^M_{pv\_idx2} \|_2 - \| P^S_{f\_idx1} - P^S_{f\_idx2} \|_2 \right| < 2mr \quad \textbf{(6- 4)}$$

with *mr* being the scene point cloud resolution and *idx1, idx2* the vertex indices with *idx1<idx2*.

These checks repeat until all correspondences from the k-NNDR stage are grouped into one of the clusters $H_a \in \boldsymbol{H}$. Clusters that have a cardinality greater than 66% of the largest $\boldsymbol{H}$ are maintained and comprise $H_b, \{b \mid b \in \mathbb{N}, b < a\}$ with $H_b \in \boldsymbol{H'}$, while the rest are discarded as too small.

6.3.2.3 Hypothesis generation and verification

Each cluster H_b of the $\boldsymbol{H'}$ cluster set defines a transformation hypothesis T_b between the model and the target. Although these hypotheses are based on correspondences twice refined for outliers (k-NNDR matches and geometric consistency checks), some outliers may still exist that are not consistent with a unique rigid transformation i.e. 3D rotation and 3D translation of the target within the scene. Therefore inconsistent correspondences within the same transformation hypothesis T_b are discarded based on the Random Sample Consensus (RANSAC) [156] algorithm using 1000 iterations.

RANSAC randomly selects a small set of correspondences and calculates the rigid transformation that aligns the model keypoints to the scene keypoints in means of a rotation matrix *R* and a translation vector *T*. It then applies this transformation to the model keypoints, measures the distance to their corresponding scene keypoints and counts the number of inliers that are consistent with the transformation based on a threshold. For a given set of template – scene correspondences within H_b this process repeats until a least square minimization solution is found or the maximum number of iterations are reached:

$$\text{argmin}_{R,T} = \sum_{idx=1}^{w} \left\| P^{S}_{f\_idx} - R \cdot P^{M}_{pv\_idx} - T \right\|_2^2 \qquad \textbf{(6- 5)}$$

with *w* being the cardinality of the cluster H_b.

Finally, a geometrical cue verification task rejects false transformation hypotheses by applying each T_b on the ideal 3D point cloud model $\boldsymbol{P_{mt}}$ creating $\boldsymbol{P_{mf}}$. The latter is fine aligned with the scene point cloud $\boldsymbol{P_f}$ via the Iterative Closest Point (ICP) [208] technique using 1000 iterations. Finally, a hypothesis H_b is accepted based on the Hypothesis verification pseudo code shown in Algorithm 6-1. The enhanced 3D ATR pipeline is shown in Figure 6- 5. Since HoD and HoD-S and HoD require online scene resolution estimation and template keypoint description for each individual scene, the 3D ATR pipeline is accordingly modified to facilitate these requirements (Figure 6- 6).

Algorithm 6- 1 Hypothesis Verification Pseudo-code

```
1  function Hypothesis Verification
   Input: transformed model Pmf & scene Pf point clouds
   Output: 100*(N/T) %
2  For each aligned model point Pmf
3      find the nearest scene neighbour
4      Count N= number of points with a squared nearest
                neighbour distance < 2mr
5      Count T= total number of aligned model points
6  end
7  Accept hypothesis if Output > 1%
```

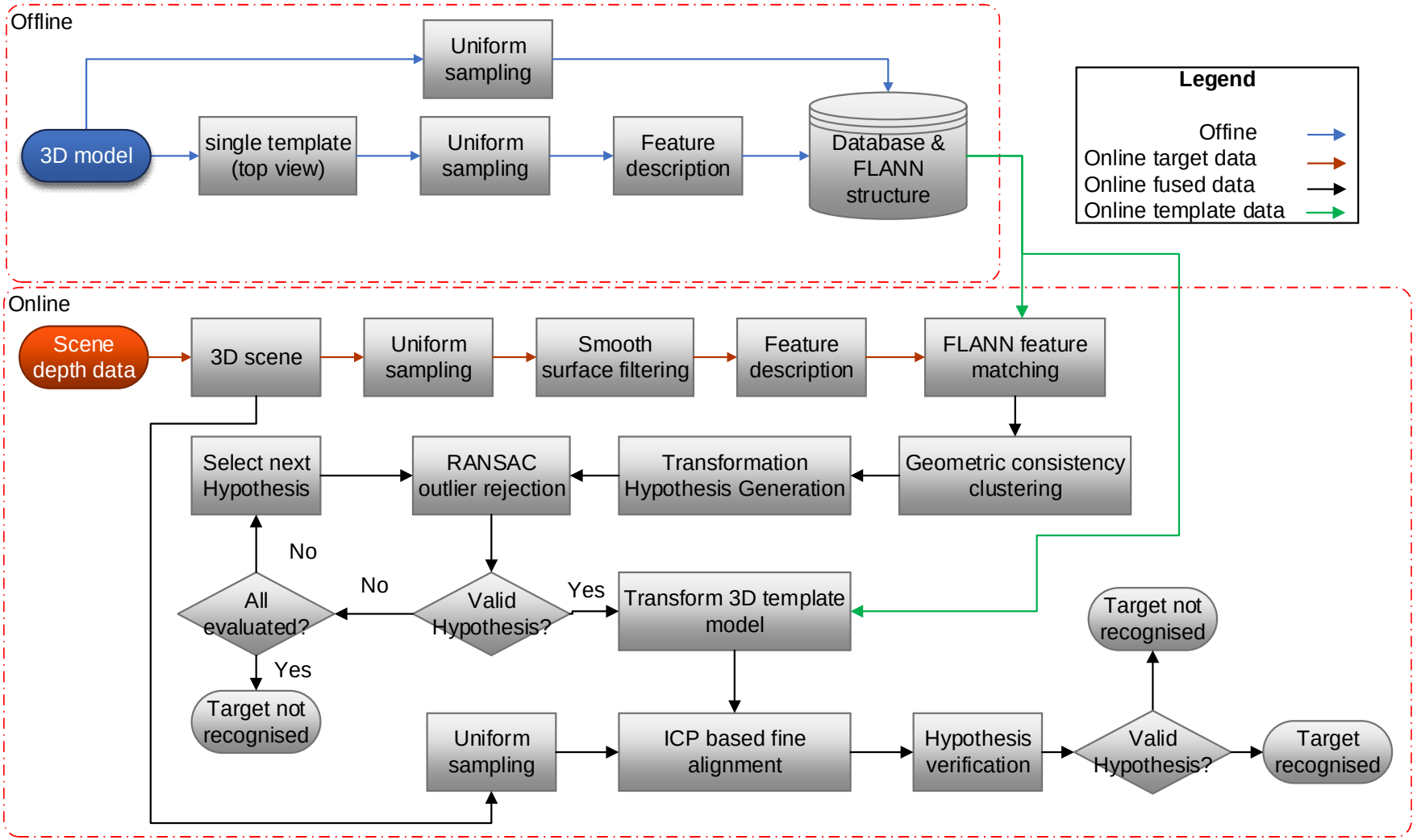


Figure 6- 5 Typical 3D ATR pipeline for the computer vision based 3D local descriptors (image from [23])

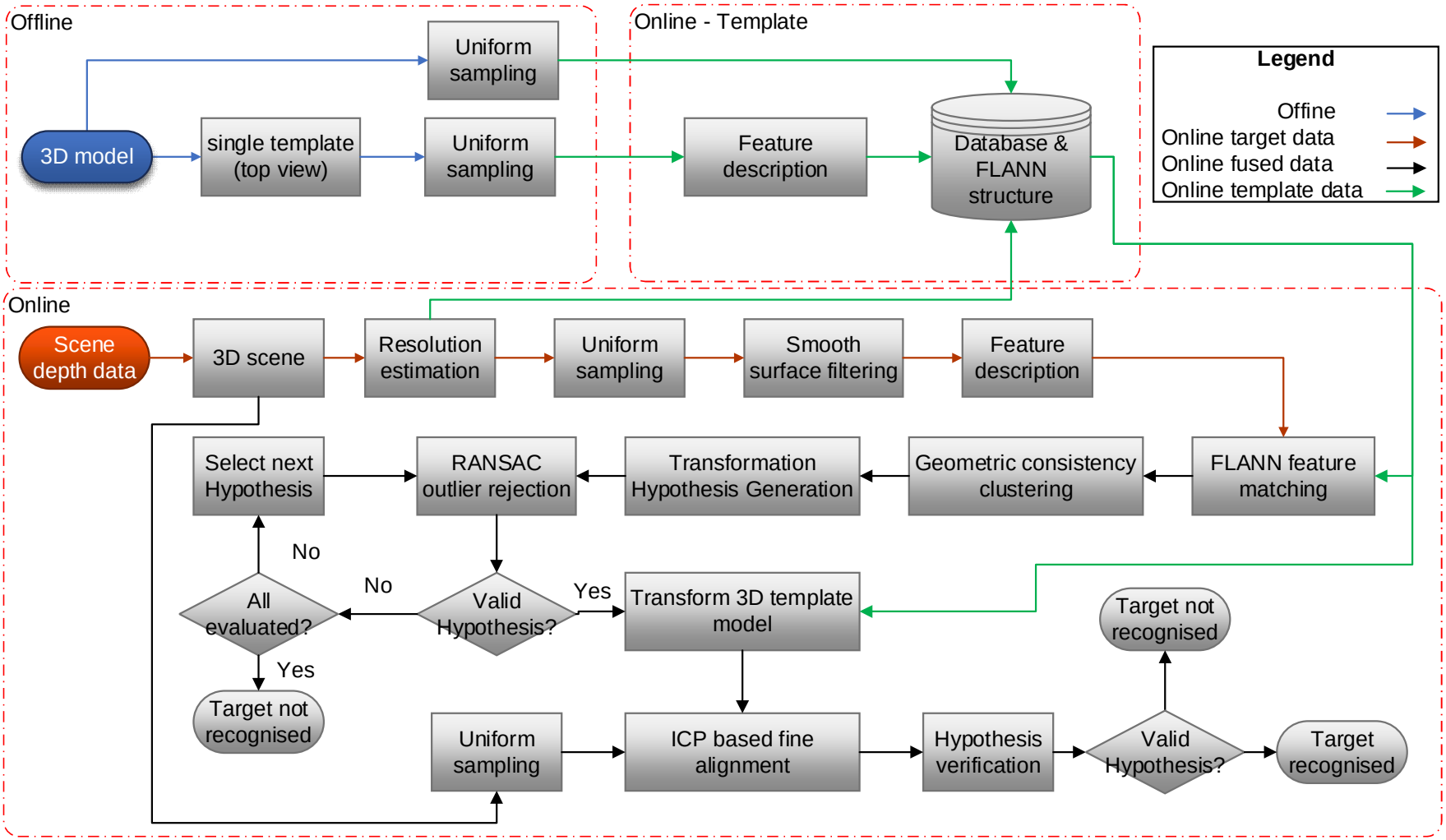


Figure 6- 6 3D ATR pipeline for the HoD and HoD-S (image from [23])

6.4 Experimental Setup

6.4.1 Synthetic engagement scenarios

Real laser scans of military scenarios are restricted and thus the OpenFlight simulation software [209] is used to simulate the highly credible air-to-ground missile engagement scenarios presented in Table 6-1. OpenFlight is a broadly used highly realistic real-time visualization simulation software capable of creating accurate depth image sceneries such as 2.5D scene images. These sceneries can then be converted into 3D point clouds simulating active laser fingerprints which can be further exploited to perform ATR. Models include both military and non-military objects to support the creation of realistic scenarios.

Through OpenFlight, three scenarios are simulated in which an ATR capable LIDAR based missile with ATR capabilities observes a ground based environment. Each scenario includes several runs resulting in a total of 787 scenes. Scenarios involve a rural context while the missile is flying at varying altitudes and headings, under various pitch, roll and yaw angles while in parallel the missile is at several distances from a stationary Leopard C2 A1 MEXAS MBT (Figure 6- 7). The difficulty of each scenario is further raised by increasing the amount of occlusion and clutter (non-target objects) such as buildings or trees. Additionally, artificial sensor noise and atmospheric disturbances are added to make the tests even more challenging. Compared to [9]–[11], [13], [71], [75], these scenarios can be considered as more realistic and challenging since they are affected by a greater number of parameters with the most significant ones being the missile-target range related effects.

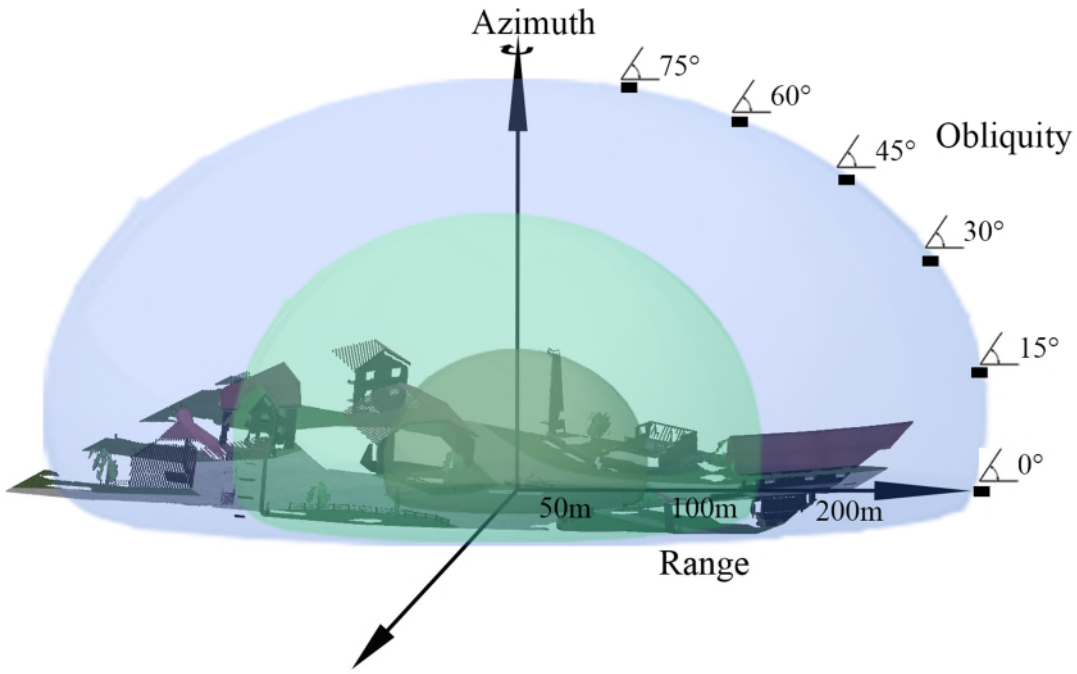


Figure 6- 7 3D point cloud construction with scenario variables shown (texture is only for better representation purposes)

Table 6- 1 Parameters per scenario

Scenario	1	2	3
Runs	6	6	1
Obliquity (°)	0°–75° per 15°	0°–75° per 15°	30°
Range (m)	50	100	200
Nº of scenes / with target	345/ 334	364/ 327	78/ 78

6.4.2 Evaluation criteria

Recognition performance is evaluated based on the confusion matrix of Table 6-2 with the 2-meter target – model distance threshold being linked to the effective miss-distance range of a missile. Currently, 2D and 3D descriptors are evaluated based on a precision-recall curve [52], [59], [63], [89], [90], [112], [114], [206], [210]. In this evaluation though, this is not possible as the ATR architecture does not exploit a variable NNDR threshold but rather rejects matches based on

Geometric consistency checks. In addition, the *TN* case is not always applicable because in some runs the target is always present. Therefore, a suitable performance metric is the F1-score which encapsulates both precision and recall information in a single value neglecting *TN*:

$$F1-score=\frac{2\#TP}{2\#TP+\#FP+\#FN} \quad \textbf{(6- 6)}$$

where # denotes the number of the entity that follows.

Table 6- 2 Confusion matrix

		Predicted	
		Yes	No
Actual	Yes	***TP*** (True Positive) Scene has a target and $\left\|centroid(\boldsymbol{P}_{mf})-centroid(\boldsymbol{P}_{f})\right\|\le 2meters$	***FN*** (False Negative) Scene has a target but no hypothesis ***H*** is created
	No	***FP*** (False Positive) Scene does not have a target or has a target and $\left\|centroid(\boldsymbol{P}_{mf})-centroid(\boldsymbol{P}_{f})\right\|>2meters$	***TN*** (True Negative) Scene does not have a target and no hypothesis ***H*** is created

6.5 Experiments

One of the most interesting features of these trials is the increasing missile-target distance. In contrast to current 3D descriptor evaluations as in [90], [195], [206] or to purely military oriented 3D ATR manuscripts in [9]–[11], [74], the performance of the descriptors under a variable missile-target distance is investigated. Further in contrast to the contemporary literature that correlates

scale with the radius of the local volume under description [87], [165] this research refers to the distance related meaning of the term *scale*. This is important because as the missile-target distance increases, the spot size of the laser beam on the target increases and so the target's features are averaged. Depending on the missile-target range, this averaging procedure can reduce the amount of distinct features on the target considerably. However, this effect is more evident at longer distances and not at close ranges that the computer vision community is mostly interested in. An example is shown in Figure 6- 8.

The 3D ATR pipeline is in MATLAB while descriptors are implemented either in MATLAB or in C++/PCL [200] using a MEX wrapper. The parameters of each descriptor are fixed either to the ones originally proposed by their authors or to their PCL implementation [21], [87], [90], [206]. The description radius per feature descriptor is 15mr. Noise, subsampling, and the detailed performance metrics are investigated only for the benchmark scenario 3, whereas the scale invariance trials consider all three scenarios. This is because the extended missile-target range affects robustness to perturbations and therefore makes these trials even more challenging.

6.5.1.1 Scenario 1

Scenario 1 considers a 50-meter missile-target range. All descriptors evaluated excel at 15°-75° obliquity because the target pose provides distinctive details for description. However, in the 0° obliquity case (side view), the MBT's features lack of distinctiveness and clutter objects interfere with the missile-target Line-Of-Sight. Despite that, HoD-S, FPFH, 3DSC and RoPS still achieve a close to 0.9 F1-score, while the remaining competitors reach up to 0.72 (Figure 6- 9 (a)). The lowest performance is provided by SHOT and USC due to their lowest *TP* and highest *FP* scores that negatively affect the F1-score. Since these two descriptors share the same LRF, it is evident that this LRF estimation method is not robust for scenario 1.

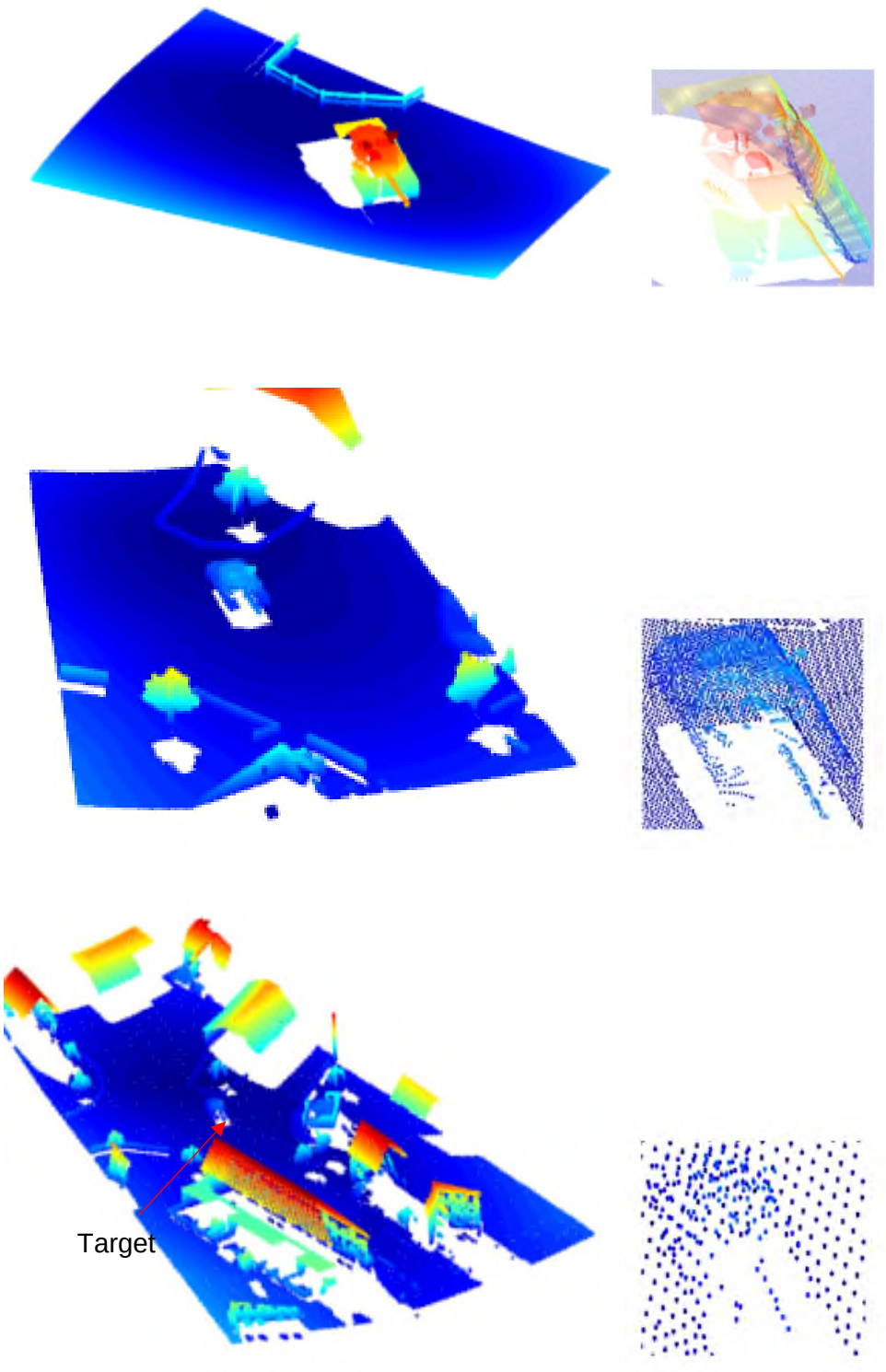


Figure 6- 8 (left column) Example scenes of scenarios 1-3 simulating distance related scene resolution (right column) corresponding MBT point cloud patch extract (image from [23]**)**

The main conclusion from this scenario, which is common for all descriptors, is that the lower the obliquity, the poorer would be the recognition performance. This happens as a result of an increase in obliquity which further causes revelation of greater part of the MBT's top-down view offering more distinct features to be encoded by the descriptors.

6.5.1.2 Scenario 2

Scenario 2 doubles the missile-target range to 100-meters. This scenario is challenging, as the size of all objects in the scene and the resolution are half the ones in scenario 1. In addition, clutter objects interfere during the 3D local descriptor phase affecting the entire ATR performance. For the 15°-75° obliquity, HoD-S, HoD and 3DSC excel with the rest of the descriptors closely following. In contrast, for the 0° case, the scale and resolution of scenario 2 combined with occlusion and the non-distinctive MBT's features at that view, impose a vast ATR performance drop for all descriptors (Figure 6- 9 (b)).

Several observations can be made from scenarios 1 and 2;

a. As the obliquity reduces, the recognition problem becomes more challenging. This can be explained as the side views of the MBT target have less distinct features compared to the top-down views.

b. Doubling the missile-target range affects the resolution of the scene depth image created. This happens because as range increases, the spot size of the laser beam increases averaging the scene's details. Hence, the resolution of the 2.5D range image is half the original and some of the target's details are not distinguishable anymore.

c. Clutter and occlusions increase as a result of the seeker's FoV capturing a greater part of the entire scene. (image from

6.5.1.3 Scenario 3

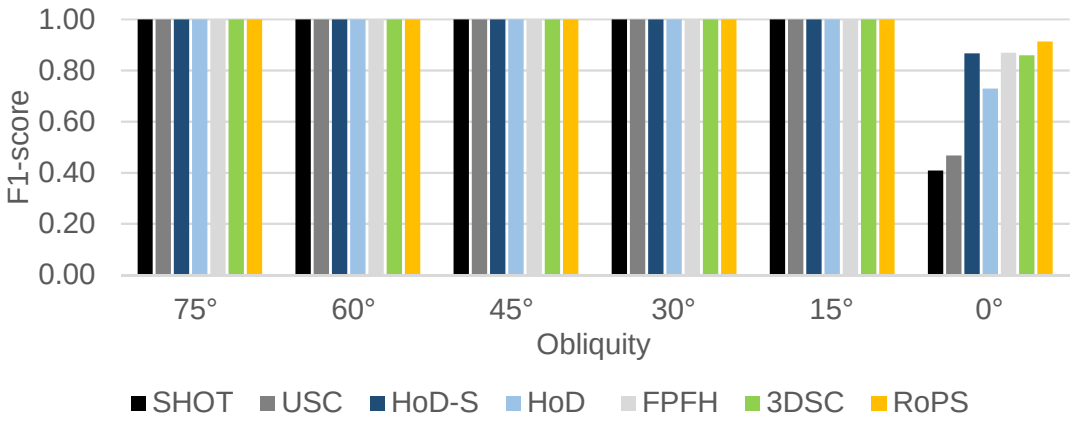
Scenario 3, considers a 200-meter missile-target range at 30° obliquity. This scenario is even more challenging as the missile-target range has quadrupled, further affecting the scene's size and resolution. Despite that, HoD-S and HoD

are quite stable delivering a fair 0.95 F1-score while 3DSC attains a 0.9. A common feature of all these descriptors is to neglect a LRF. This confirms [112] stating that it is very challenging to establish a robust LRF, and thus it is concluded that neglecting it can be beneficial as long as all potential orientation combinations are used during keypoint description. Although SHOT and USC share the same LRF, the latter performs better because it relies on a weighted sum of vertices that are located in each of its description grids, rather than relying on the prone normal variation within each description grid. Detailed results are presented in Figure 6- 9 (c).

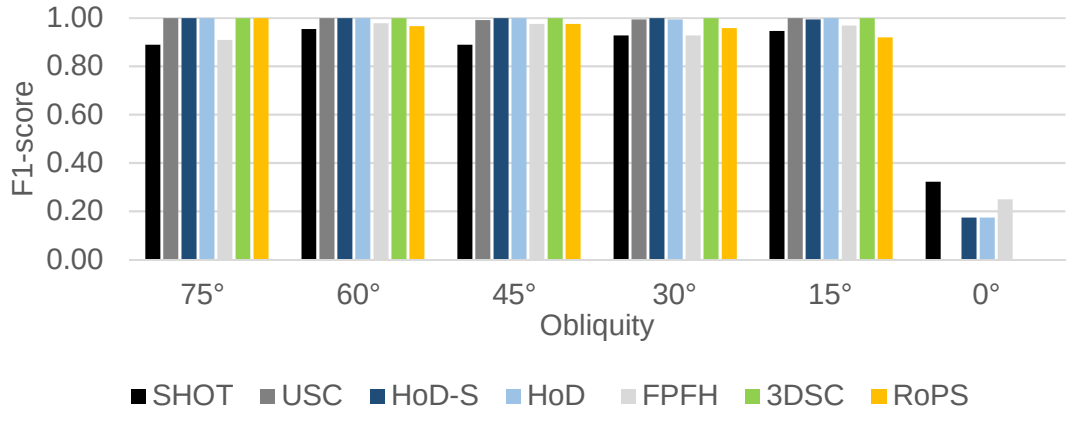
Focusing on the relationship between missile-target distance (that includes the combined scale and resolution change on the scene point cloud) and recognition performance, the following comments can be made for scenarios 1-3 and 30˚ obliquity runs:

a. HoD, HoD-S and 3DSC are the most robust to scale-change as they have only a minor performance drop. For the former two, this is due to their dynamically changing description radius that is adjusted to accommodate the requirements of each individual scene. The latter relies on a weighted count of local vertices and combined with a multi-azimuthal description scheme, it can afford scale invariance.

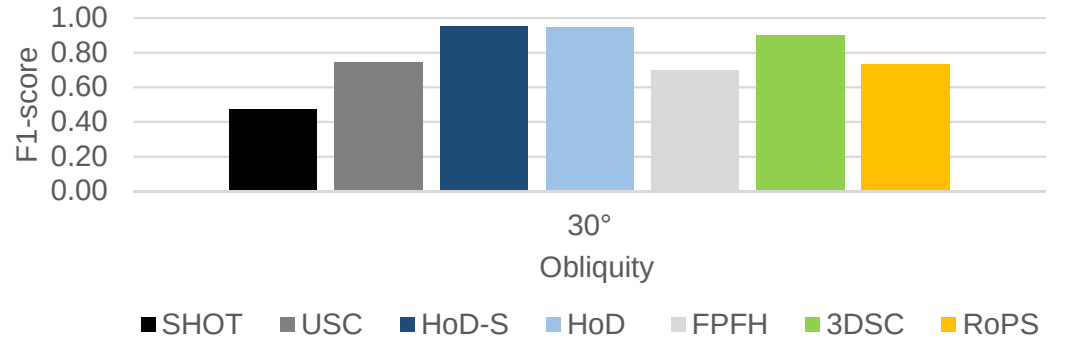
b. SHOT has a significant performance drop when the missile-target range exceeds 100 meters. This is because SHOT's descriptor solely relies on the normal variation within its description bins, with each normal vector being heavily affected by the combined subsampling and scale change of scenario 3.



(a)



(b)



(c)

Figure 6- 9 ATR performance on scenarios 1-3

6.5.1.4 Performance metrics

6.5.1.4.1 Processing efficiency

This parameter is one of the most important for time-critical missile oriented applications for the following reasons:

a. Due to the high speed of the missile, the ATR algorithm must make correct yet fast decisions so that the missile steers on time towards the target.

b. It is very important for the ATR to continuously reconfirm recognition to enhance the overall performance and reduce *FP*.

For better readability of the current Chapter, the processing efficiency trial along with all the following performance metrics are investigated only on Scenario 3 of the single template scheme. The reasoning behind selecting scenario 3 is the combination of extended missile-target range and missile obliquity. These elements comprise a common situation for a missile engagement scenario.

Despite being implemented in MATLAB, the fastest 3D descriptor generated is the HoD-S (6ms/ keypoint) due to overriding a LRF/A and having a small description size. Next are HoD (19ms/ keypoint) and FPFH (25ms/ keypoint) as the former neglects a LRF/A while the latter has the smallest descriptor size among all competitors. Although HoD encrypts the local volume twice (in a coarse and a fine manner) and is implemented in MATLAB, it is still faster compared to the FPFH which relies on a LRF and is implemented in C++. The least processing efficient 3D descriptors are USC (C++), 3DSC (C++) and RoPS (MATLAB). Compared to the most efficient HoD-S, the 3DSC is x34 slower and RoPS x44. For the RoPS specifically, it is the least efficient descriptor due to its complex LRF/A algorithm and MATLAB implementation. The processing time per descriptor is shown in Figure 6-10 (a) and per scene in Figure 6-10 (b). The latter includes the entire ATR pipeline with the common modules requiring an average of 1268ms. The relative ratios between these two figures differ as each descriptor produces a different number of kNNDR matches affecting the number of Hypotheses to be tested and thus the overall processing time. The computational breakdown of the common procedures is presented in Figure 6-10 (c) showing

that Hypothesis Verification imposes the vast computational burden because of the RANSAC and ICP iterative processes.

6.5.1.4.2 Matching accuracy

Although the 2-meter distance threshold that defines a *TP* is sufficient for the examined scenarios, descriptors achieving a translational error less than two meters can facilitate missile pinpoint targeting. Trials on Scenario 3, show that RoPS affords the smallest average error, closely followed by FPFH and USC. This is due to their LRF, provided that they positively recognise the target. Since USC is LRF based while 3DSC is LRA, the former's high accuracy is reasonable. Although the HoD-S produces the largest error, yet it remains well below 0.5-meters (Figure 6-10 (d)). Matching accuracy is calculated not only on the *TP*s but as an average value of both *TP*s and *FP*s.

6.5.1.4.3 Compactness

This metric reveals the descriptive power per element of the descriptor vector. Computer vision literature calculates compactness as the fraction of the area under curve (AUC) of the precision - recall plot divided by the number of elements that each descriptor has (float) [57]. AUC requires the number of *TN* rejections, which, in the case of benchmark scenario 3, is a non-existing case as all scene point clouds contain the target. Therefore, compactness is defined as:

$$compactness = \frac{F1-score}{float} \qquad \textbf{(6- 7)}$$

Highest compactness is achieved by HoD-S followed by FPFH. This happens as both descriptors combine a large F1-score and a small descriptor length. Lowest compactness is achieved by USC and 3DSC because even though they perform well, their float is considerably larger compared to the rest of the descriptors evaluated and thus their compactness drops dramatically (Figure 6-10 (e)). A conclusion that can be drawn is that HoD-S is an appealing descriptor for time-critical applications with hardware constraints as it combines high descriptiveness with a small float. The latter implies reduced storage memory requirements.

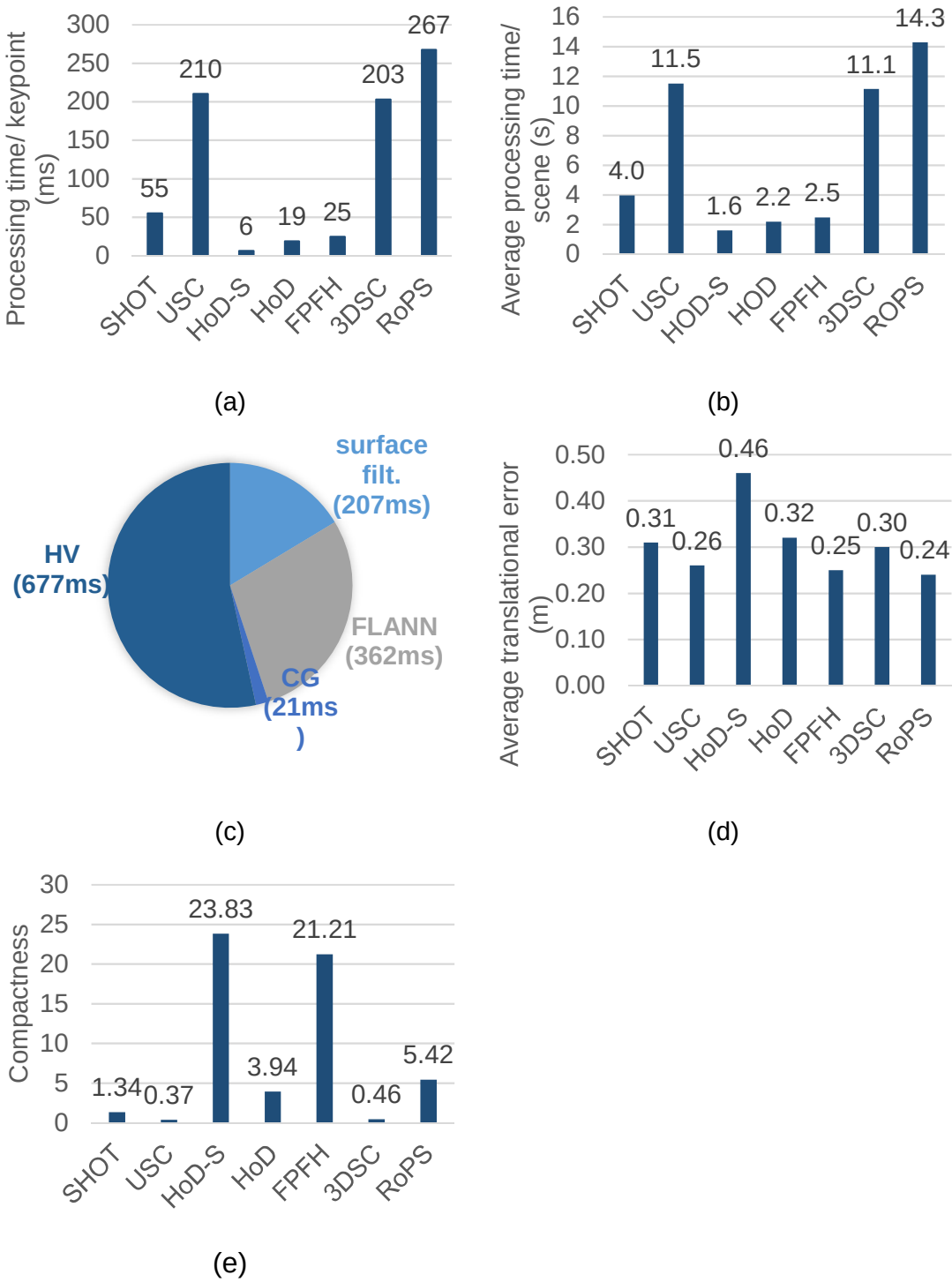


Figure 6- 10 Performance metrics (a) processing efficiency (b) average processing time per scene per descriptor (c) computational breakdown in milliseconds excluding description time (CG corresponds to Correspondence Grouping and HV to Hypotheses Verification) (d) translational error (e) compactness

6.5.1.5 Robustness to perturbations

For the purpose of this research's applications, robustness to noise and subsampling is mandatory as these perturbations are quite common in real military scenarios. Therefore, this section evaluates the performance of the 3D descriptors considered in the previous trials under several Gaussian noise and non-uniform subsampling levels applied to scenario 3. As a reminder, scenario 3 involves already subsampled scenes in order to simulate the laser spot size in relation to the missile-target range. Examples of distorted scenes are presented in Figure 6-11.

6.5.1.5.1 Sensor noise

In scenario 3 the robustness of each descriptor to Gaussian noise with zero mean and $\sigma = \{10, 20, 30\}$ cm is investigated (Figure 6- 12). The first experiment concerns $\sigma = 10$ cm where HoD, HoD-S, 3DSC, USC, RoPS and SHOT are almost unaffected. Although SHOT is a well performing 3D descriptor, its poor performance is more related to its reduced invariance to the combined scale and resolution change of scenario 3, rather than in the noise level itself. The next trial doubles the noise level, where both HoD variants, USC and 3DSC are still lightly affected. In contrast, SHOT and RoPS exceed their noise invariance limits and therefore perform quite poorly. Finally, noise triples to $\sigma = 30$ cm where all descriptors exceed their robustness capabilities. Despite that, HoD still gains a 0.5 F1-score. Detailed results per descriptor are presented in Figure 6- 13.

Regarding FPFH and its performance in all noise trials, it can be inferred that its LRF was vastly affected at all noise levels examined, confirming claims in [165].

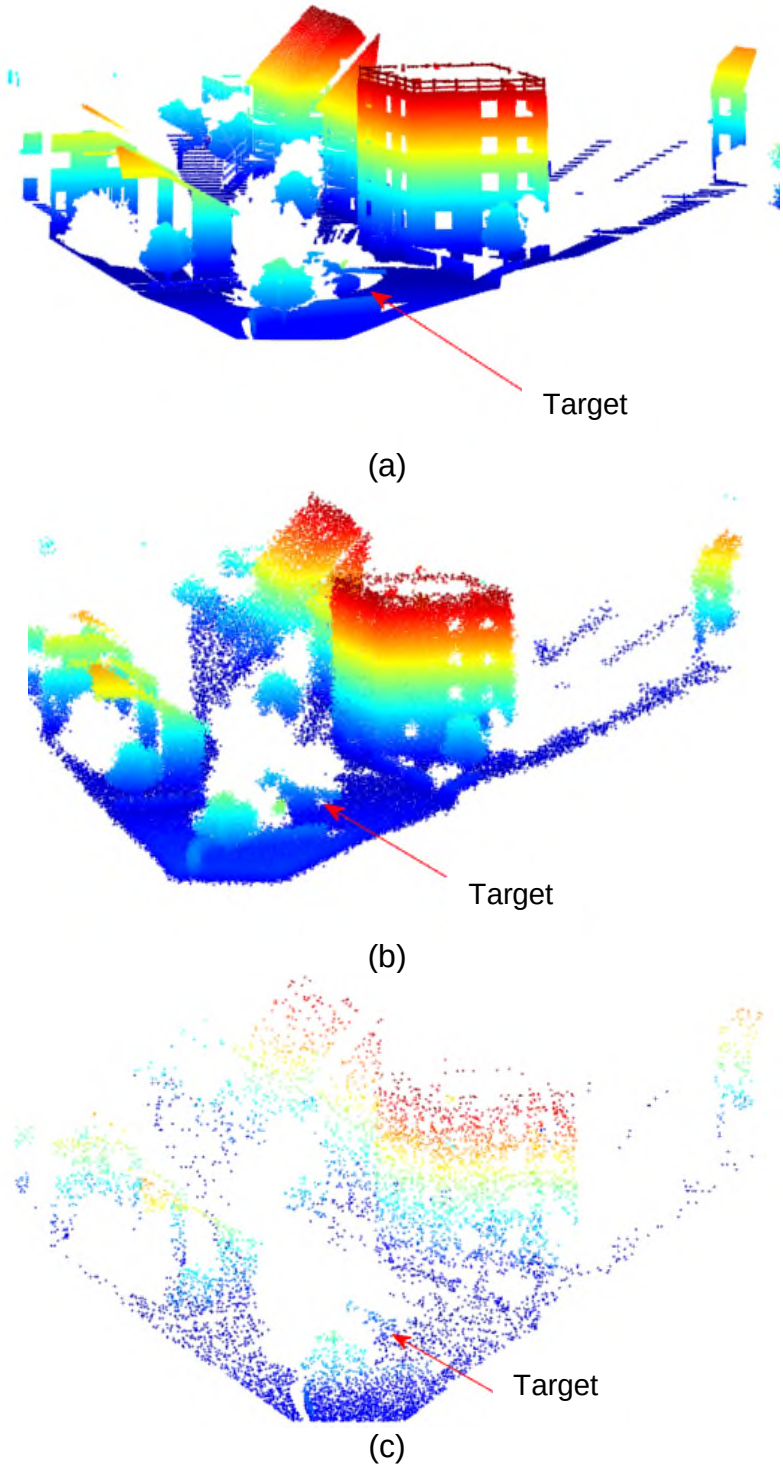


Figure 6- 11 Example a point cloud scene (a) laser spot size resolution (b) σ=20cm Gaussian noise (c) 1/16 non-uniform subsampling (image from [23]**)**

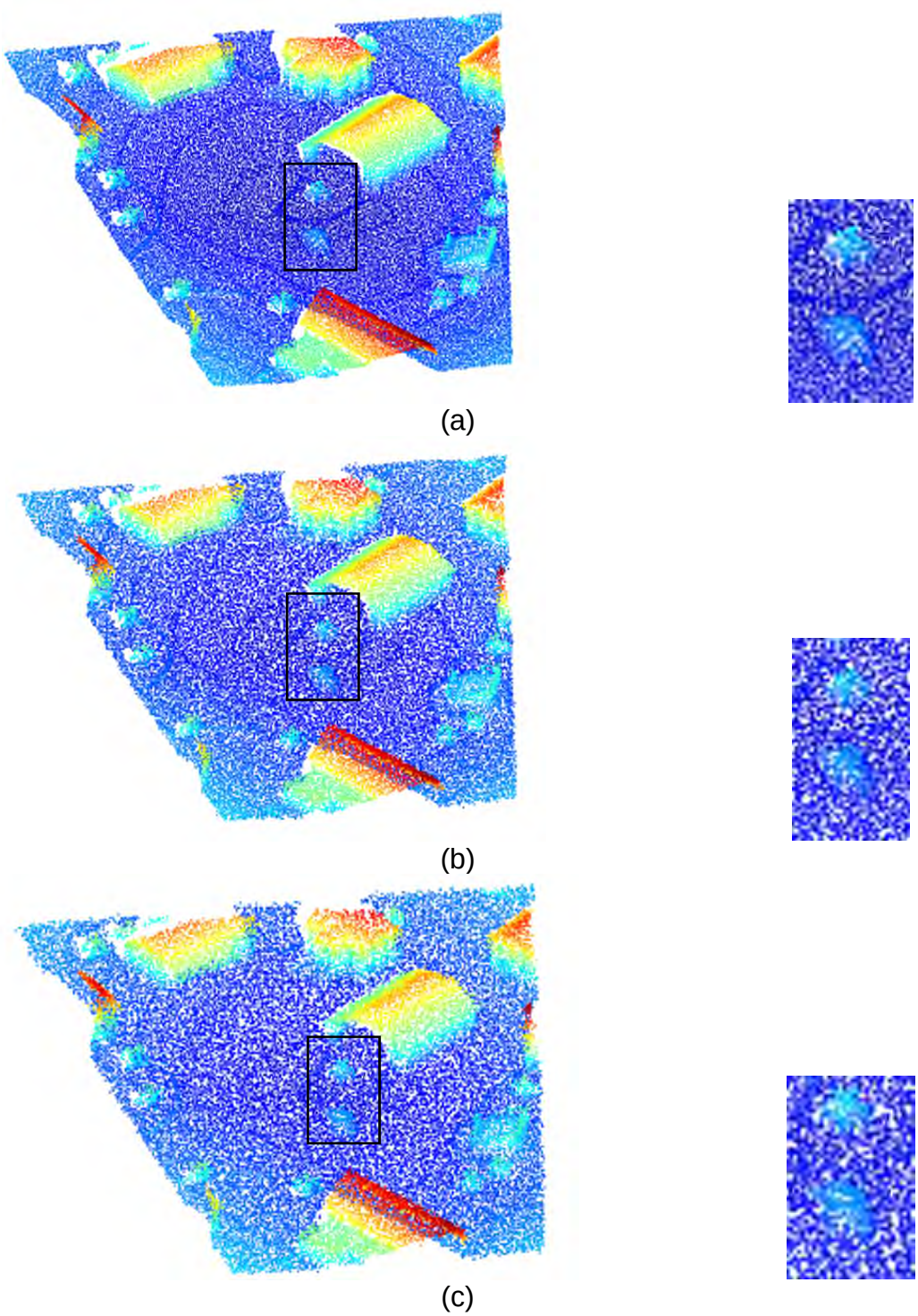


Figure 6- 12 Example of a point cloud scene under Gaussian noise with (a) σ=10cm (b) σ=20cm (c) σ=30cm (target region is zoomed in right column)

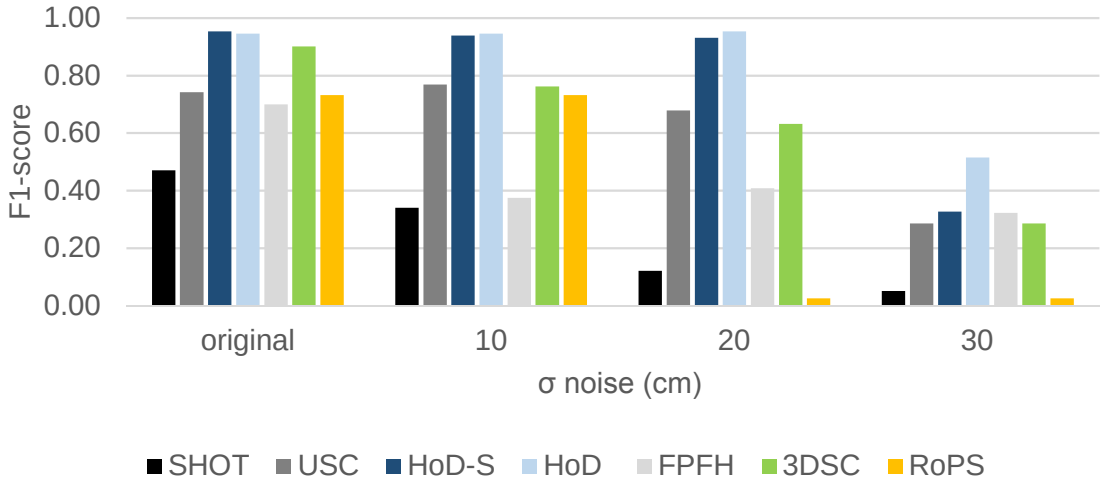
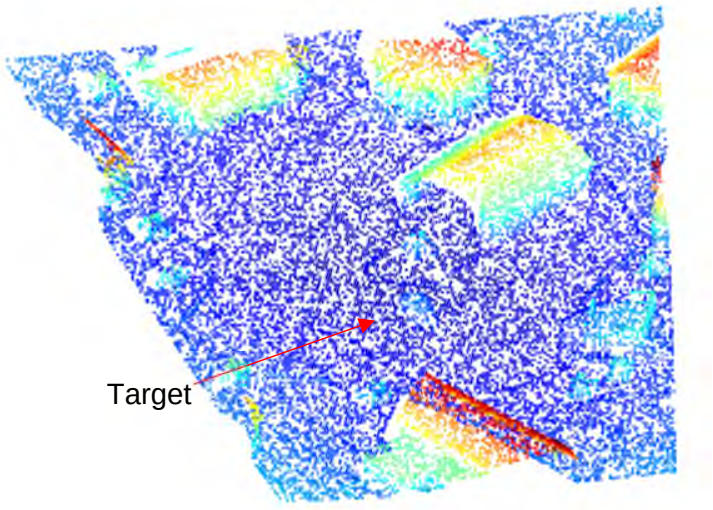


Figure 6- 13 Robustness to various noise levels

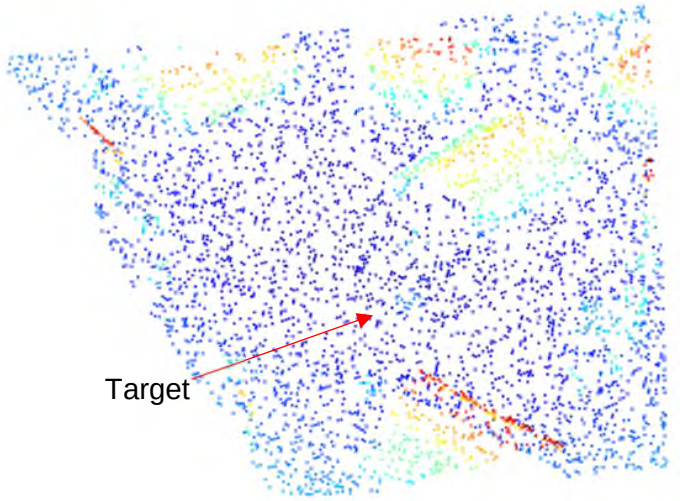
6.5.1.5.2 Non-uniform subsampling

This trial simulates the laser beam wandering and scintillation atmospheric interferences [211]. Both these effects reduce the number of reflected laser beams irregularly and randomly, and thus affect the spatial location of the scene vertices. Therefore, trials involve non-uniform and random subsampling of the scene point clouds of scenario 3 to $\{1/2, 1/8, 1/16\}$ the original scene point cloud ***P*** Figure 6- 14. This is unique, as the current literature considers only the uniform case at constant scale [63], [90], [112].

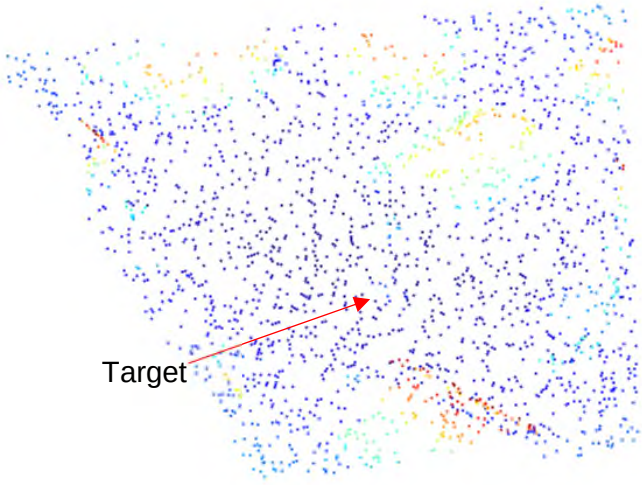
For the $1/2$ case, HoD and HoD-S are the most robust followed by 3DSC due to omitting the prone to perturbations LRF [112]. For the $1/8$ case, all competitors except HoD and HoD-S do not achieve of an appealing performance because of the simultaneous combination of scale, resolution, and subsampling-change which exceeds the limits of a repeatable LRF/A. For the $1/16$ case, all competitors fail as the combined subsampling and scale change are quite excessive. Despite that, it is worth noting that both variants of HoD are the only ones with F1-score close to 0.5 while the rest are in the order of 0.1. Figure 6-15 shows the detailed subsampling results.



(a)



(b)



(c)

Figure 6- 14 Example of a point cloud scene under non-uniform subsampling (a) $1/2$ **(b)** $1/8$ **(c)** $1/16$

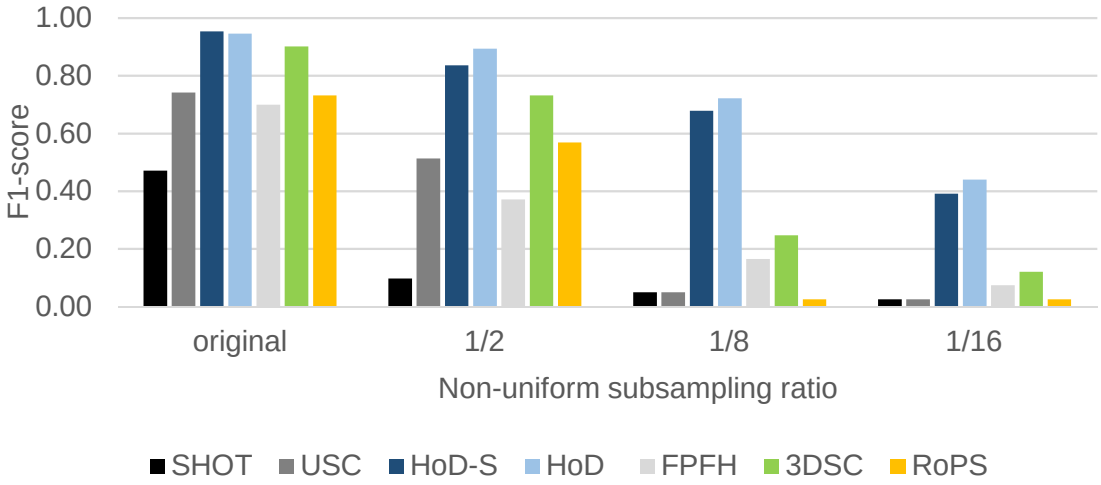


Figure 6- 15 Robustness to various non-uniform subsampling levels

6.5.1.6 Overall assessment

Figure 6-16 presents the average performance over all three scenarios and all perturbations examined. Average performance per descriptor is based on a single template scheme. Figure 6-16 also depicts the corresponding computational requirement of each descriptor. Highest performance is delivered by HoD closely followed by HoD-S as they eliminate a LRF/A, and thus offer a more stable local encoding in such an extended combination of disturbances. This conclusion is enforced by the fact that the second best performing descriptor is 3DSC that relies on an LRA. From the computational aspect, both HoD variants along with FPFH are the most efficient. The performance of each descriptor is explained as:

a. FPFH: Its LRF relies on the immediate neighbours of each keypoint and therefore is prone to disturbances confirming [90].
b. RoPS: Current trials combine disturbances that exceeded the invariance limits of its LRF, confirming that it is very challenging to establish a robust LRF.
c. USC, 3DSC: These descriptors are robust because their description process relies on a normalised weighted point-count per description bin

rather than any kind of angular based encryption that is prone to disturbances.

d. SHOT: It involves normal estimation based on small groups of points within the description radius and therefore SHOT is prone to the large disturbances investigated in this chapter.

e. HoD, HoD-S: The normalised histogram of the point-pair distances encoded can withstand large and combined nuisances due to a dynamically changing description radius that is matched to the requirements of each scene.

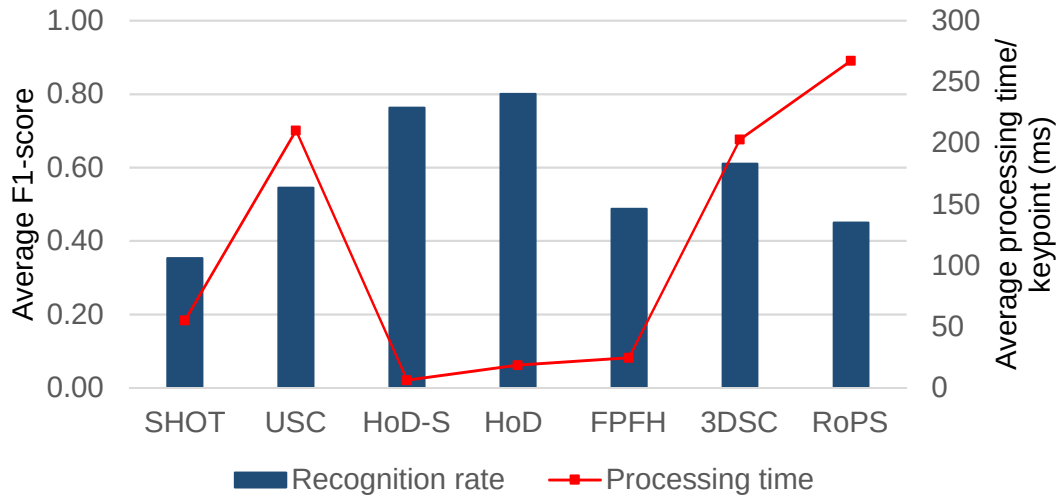


Figure 6- 16 Overall performance

6.6 Conclusion

In this chapter, current 3D descriptors are evaluated in the context of future LIDAR missile applications with ATR capabilities. This evaluation is unique as current surveys are constrained to computer vision applications and are challenged over standard commercial datasets.

Experiments are conducted exploiting a proposed pipeline that is suitable for target recognition on synthetic but highly realistic and credible scenarios. Trials involve the cases of a single template scheme while the missile is flying at various altitudes, obliquities, distances to the target under 6-DoF motion and the laser beam affected by atmospheric perturbations. Under these circumstances, all 3D

descriptors are evaluated for their processing efficiency, matching accuracy, compactness, and their robustness to atmospheric effects.

Overall, HoD and HoD-S provide an appealing solution for time-critical missile based ATR as they combine a high-performance ATR at reduced processing time. Despite HoD and HoD-S being fast to execute, future work could further improve time efficiency to accommodate this approach for high-speed missile applications in which the requirement in terms of processing time is more demanding when compared to the applications investigated in this thesis.

7 Conclusion

7.1 Overview

This thesis investigates the exploitation of computer vision algorithms for 3D Automatic Target recognition for future missile platforms. Despite the already available object recognition solutions in the computer vision community, the contemporary state-of-the-art 3D approaches fail to offer an optimum solution for time-critical military applications. The primary challenges are the processing time and hardware constraints of missile systems that must operate within a dynamically changing and highly convoluted environment.

In order to address these challenges, this research suggests a few 3D descriptors that are faster to execute compared to the ones currently available. This advantage, in addition to the efficient and appealing ATR performance, will provide insights for further research towards the future LIDAR based missile seekers. This chapter draws some conclusions for each of the contributions made in this thesis, and also suggests potential avenues for additional research. Although this doctoral research aims at the development of lightweight 3D descriptors for future intelligent missile systems meeting the missile platform constraints, the concepts presented are applicable to a variety of non-military time-critical 3D object recognition applications. Indicatively, the proposed 3D descriptors and ATR architectures are appropriate for a great range of time-critical complex systems for space, air and ground environments for military, law-enforcement and general research purposes.

7.2 Summary and discussion of contributions

The first contribution of this thesis is introduced in Chapter 2 that demonstrates a complete 3D descriptor taxonomy for each of the main descriptor types, namely one for the Local and one for the Global. Regarding the Local taxonomy, this thesis contributes in amending the existing one with details regarding the data origin and typical pre-processing stages required. This extension is important as it may reveal the extra computational burden implied by each processing phase; from the raw 3D data obtained by the LIDAR device up to the final 3D descriptor. In addition, this thesis fills the corresponding gap within the Global 3D description domain as current literature does not suggest any taxonomy. For homogeneity, both suggested taxonomies share the same philosophy.

The second contribution of this thesis (Chapter 3) is a 2.5D range image based descriptor that introduces a 3D to multi-2D problem-solving concept in conjunction with a current state-of-the-art 2D object recognition algorithm. This descriptor entitled the SURF Projection Recognition (SPR) is a complete 3D ATR solution that meets the recognition, computational, and storage memory constraints of a missile platform. It has the capability of target recognition in inter-class, intra-class and complex multi-target type scenarios. In the challenging intra-class ATR problem that consists of five military targets, four of which are highly similar, recognition exceeds 90% under several noise and subsampling disturbances that are always combined with 3D target rotation. In a limited number of complex battlefield scenarios, SPR managed to detect and recognise all targets even in the case of a multi target problem. In all instances the computational time was within the missile processing limits.

The third contribution of this research (Chapter 4) is a 3D Global based descriptor that is based on the 3D to multi-2D projection type solution followed by a global statistical analysis of the target. Ultimately, this suggestion aims at minimising the established cost function between the templates and the target. This minimisation procedure is accompanied by a CFAR based strategy such as to compensate out-of-plane rotations. Highlights of this method are its computational efficiency and the ambitious single 3D template matching scheme. A limitation of this

contribution in its current form is the requirement of either segmenting the target from the scene or implementing this algorithm to ground-to-air or air-to-air engagement scenarios. In case of the latter, the target can be segmented from the background in a relative easy manner.

The fourth contribution (Chapter 5) is a set of 3D local based descriptors. The core contribution is the Histogram of Distances (HoD) descriptor which is the basis of all offered descriptor versions. HoD has the unique features of LRF/A independence combined with a multi-encoding and multi-feature matching policy. Both attributes balance the descriptiveness and the robustness of the proposed approach to perturbations like noise and subsampling even under clutter and occlusion. The simplistic but efficient 3D point cloud encoding that neglects a computationally deficient LRF/A is the basis for an overall processing speed up. This chapter introduces two floating-point variants of HoD with different descriptor sizes under the name of HoD and HoD-S. Both alternatives offer a notable high performance in an appealing small execution time suggesting a promising solution for time-critical 3D object recognition applications. The third 3D local descriptor suggested, is the binary variant of the HoD, named the B-HoD. This descriptor has an even smaller execution time and even a further smaller storage memory requirement due to its binary nature. All three variants are challenged on commercially available datasets and compared against current state-of-the-art 3D local based descriptors. Trials under extreme noise and/ or non-uniform scene subsampling reveal the performance of each HoD variant outperforming the existing state-of-the-art 3D local descriptors accordingly.

The fifth contribution (Chapter 6) is a multi-staged missile architecture suitable for 3D missile based target detection and recognition. The main features of this architecture are the extreme case of a single template view, along with the efficient smooth surface filtering module. Several experiments are conducted exploiting the missile flight at various altitudes, obliquities, distances to the target under 6-DoF motion and the laser beam affected by atmospheric perturbations. Under these circumstances, all 3D descriptors are evaluated for their processing efficiency, matching accuracy, compactness and their robustness to atmospheric

effects. The sixth contribution is a missile oriented 3D ATR survey that reaveasl the appealing performance of HoD and HoD-S for future missile based ATR problems.

7.3 Future work

Research is an ongoing process and despite the appealing features that each of the suggested 3D descriptors possess, there is a lot of potential for further improvement. Although the suggested descriptors are faster to execute and offer an appealing ATR performance under various nuisances when compared to the currently available descriptors, future work must focus on improving the processing time efficiency to accommodate this approach in high-speed missile applications whose processing time constraints are much more demanding when compared to the work presented here. In addition,

a. For the contribution in the 2.5D domain, namely the SPR descriptor, future work may include extending trials to a larger variety of scenarios that are not limited to forested scenarios.
b. For the Global based solution, future work can focus on expanding the cost function to facilitate more parameters that would allow ATR in complex scenarios. Other work may involve creating a Global to Local architecture that will decompose the scene into several object clusters which will then undergo the suggested global 3D descriptor.
c. For the contribution in the 3D local based domain, all HoD variants may be implemented in C++ to fully utilise their processing efficiency.
d. Current research focused on computational efficient 3D descriptors, neglecting the contribution of a 3D keypoint detector. Future work should also involve research in computationally efficient 3D keypoint detectors that meet the requirements of future LIDAR based missiles.

REFERENCES

[1] G. J. Gray, N. Aouf, M. A. Richardson, B. Butters, and R. Walmsley, “An intelligent tracking algorithm for an imaging infrared anti-ship missile,” in *Proc. SPIE 8543, Technologies for Optical Countermeasures IX*, 2012, vol. 8543, p. 85430L–85430L.

[2] G. J. Gray, N. Aouf, M. a. Richardson, B. Butters, R. Walmsley, and E. Nicholls, “Feature-based recognition approaches for infrared anti-ship missile seekers,” *Imaging Sci. J.*, vol. 60, no. 6, pp. 305–320, Dec. 2012.

[3] S.-G. Sun, “Automatic target recognition using boundary partitioning and invariant features in forward-looking infrared images,” *Opt. Eng.*, vol. 42, no. 2, p. 524, Feb. 2003.

[4] L. Du, P. Wang, H. Liu, M. Pan, and F. Chen, “Bayesian Spatiotemporal Multitask Learning for Radar HRRP Target Recognition,” *IEEE Trans. Signal Process.*, vol. 59, no. 7, pp. 3182–3196, 2011.

[5] L. Du, H. Liu, P. Wang, B. Feng, M. Pan, and Z. Bao, “Noise Robust Radar HRRP Target Recognition Based on Multitask Factor Analysis With Small Training Data Size,” *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3546–3559, 2012.

[6] R. Paladini, M. Martorella, and F. Berizzi, “Classification of man-made targets via invariant coherency-matrix eigenvector decomposition of polarimetric SAR/ISAR images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 8, pp. 3022–3034, 2011.

[7] D. Perissin and A. Ferretti, “Urban-target recognition by means of repeated spaceborne SAR images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4043–4058, 2007.

[8] M. Martorella, E. Giusti, A. Capria, F. Berizzi, and B. Bates, “Automatic Target Recognition by Means of Polarimetric ISAR Images and Neural Networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3786–3794, Nov. 2009.

[9] S. Roy and J. Maheux, “Baseline processing pipelilne for fast automatic target detection and recognition in airborne 3D ladar imagery,” *Autom. target recogntion XXI*, vol. 8049, May 2011.

[10] A. Vasile and R. Marino, “Pose-independent automatic target detection and recognition using 3D laser radar imagery,” *Lincoln Lab. J.*, vol. 15, no. 1, pp. 61–78, 2005.

[11] X. L. Xiaofeng Li, J. X. Jun Xu, J. L. Jijun Luo, L. C. Lijia Cao, and S. Z. Shengxiu Zhang, “Ground target recognition based on imaging LADAR point cloud data,” *Chinese Opt. Lett.*, vol. 10, no. s1, pp. S11002-311005, 2012.

[12] C. Grönwall, T. Chevalier, G. Tolt, and P. Andersson, “An approach to target detection in forested scenes,” in *Laser Radar Technology and Applications XIII. Edited by Turner*, 2008, vol. 6950, p. 69500S–69500S–12.

[13] O. Kechagias-Stamatis and N. Aouf, “Fast 3D object matching with Projection Density Energy,” in *2015 23rd Mediterranean Conference on Control and Automation (MED)*, 2015, pp. 752–758.

[14] Kongsberg, “Naval Strike Missile - NSM.” [Online]. Available: http://www.kongsberg.com/en/kds/products/missilesystems/navalstrikemissile/#. [Accessed: 28-Sep-2015].

[15] G. Gray, N. Aouf, M. A. Richardson, B. Butters, R. Walmsley, and E. Nicholls, “Feature-Based Target Recognition in Infrared Images for Future Unmanned Aerial Vehicles,” *J. Battlef. Technol.*, vol. 14, no. 2, pp. 27–36, 2011.

[16] D. G. Lowe, “Object recognition from local scale-invariant features,” *Proc. Seventh IEEE Int. Conf. Comput. Vis.*, pp. 1150–1157 vol.2, 1999.

[17] V. M. Patel, N. M. Nasrabadi, and R. Chellappa, “Automatic target recognition based on simultaneous sparse representation,” in *17th IEEE International Conference on Image Processing (ICIP)*, 2010, pp. 1377–

1380.

[18] W. M. Brown and C. W. Swonger, “A Prospectus for Automatic Target Recognition,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 25, no. 3, pp. 401–410, 1989.

[19] R. M. Marino and W. R. Davis, “Jigsaw: a foliage-penetrating 3D imaging laser radar system,” *Lincoln Lab. J.*, vol. 15, no. 1, pp. 23–36, 2005.

[20] O. Kechagias-Stamatis, N. Aouf, and M. A. Richardson, “3D automatic target recognition for future LIDAR missiles,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 6, pp. 2662–2675, Dec. 2016.

[21] O. Kechagias-Stamatis and N. Aouf, “Histogram of distances for local surface description,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, vol. 2016–June, pp. 2487–2493.

[22] O. Kechagias-stamatis and N. Aouf, “B - HoD : A Lightweight and Fast Binary descriptor for 3D Object Recognition and Registration,” in *4th IEEE International Conference on Networking, Sensing and Control (ICNSC)*, 2017.

[23] O. Kechagias-Stamatis and N. Aouf, “Evaluating 3D Local Descriptors for Future LIDAR Missiles with Automatic Target Recognition Capabilities,” *Imaging Sci. J.*, 2017.

[24] O. Kechagias-Stamatis, N. Aouf, and D. Nam, “3D Automatic Target Recognition for UAV Platforms,” in *Sensor Signal Processing for Defence (SSPD2017)*, 2017.

[25] K. Berker Logoglu, S. Kalkan, and A. Temizel, “CoSPAIR: Colored Histograms of Spatial Concentric Surflet-Pairs for 3D object recognition,” *Rob. Auton. Syst.*, vol. 75, no. October, pp. 558–570, 2015.

[26] F. Alhwarin, C. Wang, D. Risti, and A. Gräser, “Improved SIFT-Features Matching for Object Recognition,” pp. 179–190, 2008.

[27] W. Wohlkinger and M. Vincze, “Ensemble of shape functions for 3D object

classification," in *2011 IEEE International Conference on Robotics and Biomimetics*, 2011, pp. 2987–2992.

[28] A. Aldoma *et al.*, "CAD-model recognition and 6DOF pose estimation using 3D cues," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 585–592.

[29] K. Brki, A. Aldomà, M. Vincze, S. Šegvi, and Z. Kalafati, "Temporal Ensemble of Shape Functions," *EurographicsWorkshop 3D Object Retr.*, 2014.

[30] Z.-C. Marton, D. Pangercic, N. Blodow, and M. Beetz, "Combined 2D-3D categorization and classification for multimodal perception systems," *Int. J. Rob. Res.*, vol. 30, no. 11, pp. 1378–1402, Sep. 2011.

[31] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "Point feature extraction on 3D range scans taking into account object boundaries," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2601–2608.

[32] A. Aldoma, F. Tombari, R. B. Rusu, and M. Vincze, "OUR-CVFH – Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7476 LNCS, 2012, pp. 113–122.

[33] A. Collignon, D. Vandermeulen, P. Suetens, and G. Marchal, "Registration of 3D multi-modality medical images using surfaces and point landmarks," *Pattern Recognit. Lett.*, vol. 15, no. 5, pp. 461–467, May 1994.

[34] S. Allaire, J. J. Kim, S. L. Breen, D. A. Jaffray, and V. Pekar, "Full orientation invariance and improved feature selectivity of 3D SIFT with application to medical image analysis," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2008, pp. 1–8.

[35] B. Zheng, R. Ishikawa, T. Oishi, J. Takamatsu, and K. Ikeuchi, "6-DOF pose

estimation from single Ultrasound image using 3D IP models," *2008 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, pp. 1–8, 2008.

[36] U. Castellani *et al.*, "A New Shape Diffusion Descriptor for Brain Classification," in *Medical Image Computing and Computer-Assisted Intervention -- MICCAI 2011: 14th International Conference, Toronto, Canada, September 18-22, 2011, Proceedings, Part II*, G. Fichtinger, A. Martel, and T. Peters, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 426–433.

[37] Jing Hua, Zhaoqiang Lai, Ming Dong, Xianfeng Gu, and Hong Qin, "Geodesic Distance-weighted Shape Vector Image Diffusion," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1643–1650, Nov. 2008.

[38] Y. Wang, B. S. Peterson, and L. H. Staib, "3D Brain surface matching based on geodesics and local geometry," *Comput. Vis. Image Underst.*, vol. 89, no. 2–3, pp. 252–271, Feb. 2003.

[39] G. Pang and U. Neumann, "Training-based object recognition in cluttered 3D point clouds," *Proc. - 2013 Int. Conf. 3D Vision, 3DV 2013*, pp. 87–94, 2013.

[40] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1252–1262, Jul. 2007.

[41] Y. Lei, H. Lai, and X. Jiang, "3D face recognition by SURF operator based on depth image," in *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010*, 2010, vol. 9, pp. 240–244.

[42] T. Hou and H. Qin, "Efficient computation of scale-space features for deformable shape correspondences," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6313 LNCS, no. PART 3, pp. 384–397, 2010.

[43] H. Chen and B. Bhanu, "Efficient recognition of highly similar 3D objects in

range images.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 1, pp. 172–9, Jan. 2009.

[44] B. Alsadik, M. Gerke, and G. Vosselman, "Visibility analysis of point cloud in close range photogrammetry," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. II-5, no. June, pp. 9–16, May 2014.

[45] Min Lu, Yulan Guo, Jun Zhang, Jianwei Wan, and J. Li, "Automatic markerless registration of mobile LiDAR point-clouds," in *2014 IEEE Geoscience and Remote Sensing Symposium*, 2014, pp. 173–176.

[46] W. Armbruster and M. Hammer, "Maritime target identification in flash-ladar imagery," in *Spie*, 2012, vol. 8391, p. 83910C–83910C–9.

[47] M. Hammer, M. Hebel, and M. Arens, "Maritime target identification in gated viewing imagery," 2015, vol. 9649, p. 96490K.

[48] W. Armbruster and M. Hammer, "Segmentation, classification, and pose estimation of maritime targets in flash-ladar imagery," in *Spie*, 2012, vol. 8542, p. 85420K.

[49] P. Bariya, J. Novatnack, G. Schwartz, and K. Nishino, "3D geometric scale variability in range images: Features and descriptors," *Int. J. Comput. Vis.*, vol. 99, no. 2, pp. 232–255, 2012.

[50] T. Darom and Y. Keller, "Scale-Invariant Features for 3-D Mesh Models," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2758–2769, May 2012.

[51] Y. Guo, F. Sohel, M. Bennamoun, J. Wan, and M. Lu, "An Accurate and Robust Range Image Registration Algorithm for 3D Object Modeling," *IEEE Trans. Multimed.*, vol. 16, no. 5, pp. 1377–1390, Aug. 2014.

[52] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," *Comput. Vision–ECCV 2010*, pp. 356–369, 2010.

[53] N. Bayramoglu and A. A. Alatan, "Shape Index SIFT: Range Image Recognition Using Local Features," in *2010 20th International Conference*

on Pattern Recognition, 2010, pp. 352–355.

[54] U. Castellani, M. Cristani, S. Fantoni, and V. Murino, “Sparse points matching by combining 3D mesh saliency with statistical descriptors,” *Comput. Graph. Forum*, vol. 27, no. 2, pp. 643–652, 2008.

[55] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, “Recognizing Objects in Range Data Using Regional Point Descriptors,” in *ECCV*, vol. 3023, 2004, pp. 224–237.

[56] Yulan Guo, F. a. Sohel, M. Bennamoun, Jianwei Wan, and Min Lu, “Integrating shape and color cues for textured 3D object recognition,” in *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, 2013, pp. 1614–1619.

[57] Yulan Guo, M. Bennamoun, F. Sohel, Min Lu, and Jianwei Wan, “3D Object Recognition in Cluttered Scenes with Local Surface Features: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2270–2287, Nov. 2014.

[58] M. Lu, Y. Guo, J. Zhang, Y. Ma, and Y. Lei, “Recognizing Objects in 3D Point Clouds with Multi-Scale Local Features,” *Sensors*, vol. 14, no. 12, pp. 24156–24173, Dec. 2014.

[59] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, “Rotational Projection Statistics for 3D Local Surface Description and Object Recognition,” *Int. J. Comput. Vis.*, vol. 105, no. 1, pp. 63–86, Oct. 2013.

[60] Yulan Guo, F. a. Sohel, M. Bennamoun, Jianwei Wan, and Min Lu, “RoPS: A local feature descriptor for 3D rigid objects based on rotational projection statistics,” in *2013 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, 2013, pp. 1–6.

[61] Y. Guo, M. Bennamoun, F. A. Sohel, J. Wan, and M. Lu, “3D free form object recognition using rotational projection statistics,” in *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, 2013, pp. 1–8.

[62] Y. Guo, F. Sohel, M. Bennamoun, J. Wan, and M. Lu, “A novel local surface feature for 3D object recognition under clutter and occlusion,” *Inf. Sci. (Ny).*, vol. 293, pp. 196–213, Feb. 2015.

[63] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, “TriSI : A Distinctive Local Surface Descriptor for 3D Modeling and Object Recognition,” in *8th International Conference on Computer Graphics Theory and Applications*, 2013.

[64] A. E. Johnson and M. Hebert, “Using spin images for efficient object recognition in cluttered 3D scenes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, 1999.

[65] A. Flint, A. Dick, and A. Van Den Hengel, “Thrift: Local 3D Structure Recognition,” in *9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications (DICTA 2007)*, 2007, pp. 182–188.

[66] Y. Gao, Q. Dai, and N.-Y. Zhang, “3D model comparison using spatial structure circular descriptor,” *Pattern Recognit.*, vol. 43, no. 3, pp. 1142–1151, 2010.

[67] G. Flitton, T. Breckon, and N. Megherbi Bouallagu, “Object Recognition using 3D SIFT in Complex CT Volumes,” *Procedings Br. Mach. Vis. Conf. 2010*, p. 11.1-11.12, 2010.

[68] G. Flitton, T. P. Breckon, and N. Megherbi, “A comparison of 3D interest point descriptors with application to airport baggage object detection in complex CT imagery,” *Pattern Recognit.*, vol. 46, no. 9, pp. 2420–2436, Sep. 2013.

[69] G. Flitton, T. P. Breckon, and N. Megherbi, “A 3D Extension to Cortex Like Mechanisms for 3D Object Class Recognition,” no. June, pp. 3634–3641, 2012.

[70] W. Armbruster, “Model-based object recognition in range imagery,” in *Proceedings of SPIE*, 2009, vol. 7481, p. 748102.

[71] W. Armbruster, “Exploiting range imagery: techniques and applications,” in *Symposium on Photoelectronic Detection and Imaging*, 2009, vol. 7382, pp. 738203-738203–12.

[72] W. Armbruster, “NATO RTO Exploiting 3D LADAR : Techniques and Applications,” pp. 1–24.

[73] D. Gibbins, “3D Target Recognition Using 3-Dimensional SIFT or Curvature key-points and Local spin Descriptors,” in *Defence Applications of Signal Processing 20092*, 2009, pp. 1–6.

[74] C. Grönwall, “Ground object recognition using laser radar data: geometric fitting, performance analysis, and applications,” Linköping, Sweden, 2006.

[75] C. Gronwall, “Ground target recognition using rectangle estimation,” *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3400–3408, 2006.

[76] L.-P. Bergé, N. Aouf, T. Duval, and G. Coppin, “Generation and VR Visualization of 3D Point Clouds for Drone Target Validation Assisted by an Operator,” in *CEEC 2016 : 8th Computer Science and Electronic Engineering Conference*, 2016.

[77] Y. Guo, J. Zhang, M. Lu, J. Wan, and Y. Ma, “Benchmark datasets for 3D computer vision,” in *2014 9th IEEE Conference on Industrial Electronics and Applications*, 2014, no. JUNE 2014, pp. 1846–1851.

[78] R. D. Richmond and S. C. Cain, *Direct-Detection LADAR Systems*, 1st ed. 1000 20th Street, Bellingham, WA 98227-0010 USA: SPIE, 2010.

[79] “Advanced Scientific Concepts.” [Online]. Available: http://www.advancedscientificconcepts.com/technology/technology.html. [Accessed: 21-Apr-2017].

[80] B. Lohani, S. Chacko, S. Ghosh, and S. Sasidharan, “Surveillance system based on Flash LiDAR,” in *Proceedings of XXXII INCA International Congress on Cartography for Sustainable Earth Resource Management, Indian Cartographer*, 2013, pp. 77–85.

[81] A. S. Mian, M. Bennamoun, and R. Owens, “Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1584–1601, Oct. 2006.

[82] Velodyne, “Puck VLP16.” [Online]. Available: http://velodynelidar.com/vlp-16.html. [Accessed: 11-Mar-2017].

[83] W. Sun, Y. Hu, D. G. MacDonnell, C. Weimer, and R. R. Baize, “Technique to separate lidar signal and sunlight,” *Opt. Express*, vol. 24, no. 12, p. 12949, 2016.

[84] C. Dorai and A. K. Jain, “COSMOS-A representation scheme for 3D free-form objects,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 10, pp. 1115–1130, Oct. 1997.

[85] Y. Zhong, “Intrinsic shape signatures: A shape descriptor for 3D object recognition,” in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, 2009, pp. 689–696.

[86] A. S. Mian, M. Bennamoun, and R. Owens, “On the Repeatability and Quality of Keypoints for Local Feature-based 3D Object Retrieval from Cluttered Scenes,” *Int. J. Comput. Vis.*, vol. 89, no. 2–3, pp. 348–361, Sep. 2009.

[87] F. Tombari, S. Salti, and L. Di Stefano, “Performance evaluation of 3D keypoint detectors,” *Int. J. Comput. Vis.*, vol. 102, no. 1–3, pp. 198–220, 2013.

[88] M. Muja and D. G. Lowe, “Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration,” *Int. Conf. Comput. Vis. Theory Appl. (VISAPP ’09)*, pp. 1–10, 2009.

[89] F. Tombari, S. Salti, and L. Di Stefano, “Unique shape context for 3d data description,” in *Proceedings of the ACM workshop on 3D object retrieval - 3DOR ’10*, 2010, p. 57.

[90] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and N. M. Kwok, “A Comprehensive Performance Evaluation of 3D Local Feature Descriptors,” *Int. J. Comput. Vis.*, vol. 116, no. 1, pp. 66–89, Jan. 2016.

[91] B. Bustos, D. A. Keim, D. Saupe, T. Schreck, and D. V. Vranić, “Feature-based similarity search in 3D object databases,” *ACM Comput. Surv.*, vol. 37, no. 4, pp. 345–387, Dec. 2005.

[92] D. G. Lowe, “Distinctive image features from scale invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[93] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-Up Robust Features (SURF),” *Comput. Vis. Image Underst.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[94] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, “BRIEF: Computing a Local Binary Descriptor very Fast.,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1281–1298, Nov. 2011.

[95] E. Rublee and G. Bradski, “ORB - an efficient alternative to SIFT or SURF,” 2011.

[96] S. Leutenegger, M. Chli, and R. Y. Siegwart, “BRISK: Binary Robust invariant scalable keypoints,” *2011 Int. Conf. Comput. Vis.*, pp. 2548–2555, Nov. 2011.

[97] a. Alahi, R. Ortiz, and P. Vandergheynst, “FREAK: Fast Retina Keypoint,” *2012 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 510–517, Jun. 2012.

[98] J. A. Whitworth, “Best practices in use of research evidence to inform health decisions.,” *Health Res. Policy Syst.*, vol. 4, no. 1, p. 11, 2006.

[99] S. Tang *et al.*, “Histogram of Oriented Normal Vectors for Object Recognition with a Depth Sensor,” in *11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part II*, 2013, pp. 525–538.

[100] E. R. Nascimento, G. L. Oliveira, M. F. M. Campos, A. W. Vieira, and W.

R. Schwartz, “BRAND: A robust appearance and depth descriptor for RGB-D images,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 1720–1726.

[101] E. R. Do Nascimento, G. L. Oliveira, A. W. Vieira, and M. F. M. Campos, “On the development of a robust, fast and lightweight keypoint descriptor,” *Neurocomputing*, vol. 120, pp. 141–155, 2013.

[102] A. Shaiek and F. Moutarde, “Fast 3D Keypoints Detector and Descriptor for View-Based 3D Objects Recognition,” in *Advances in Depth Image Analysis and Applications: International Workshop, WDIA 2012, Tsukuba, Japan, November 11, 2012, Revised Selected and Invited Papers*, X. Jiang, O. R. P. Bellon, D. Goldgof, and T. Oishi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 106–115.

[103] G. Pang and U. Neumann, “Fast and Robust Multi-view 3D Object Recognition in Point Clouds,” in *2015 International Conference on 3D Vision*, 2015, pp. 171–179.

[104] A. Aldoma, F. Tombari, L. Di Stefano, and M. Vincze, “A Global Hypotheses Verification Method for 3D Object Recognition,” in *Computer Vision--ECCV 2012*, Springer, 2012, pp. 511–524.

[105] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, “Matching 3D models with shape distributions,” in *Proceedings International Conference on Shape Modeling and Applications*, 2001, pp. 154–166.

[106] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, “Fast 3D recognition and pose using the Viewpoint Feature Histogram,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 2155–2162.

[107] Y. Salih, A. S. Malik, D. Sidibe, M. Simsim, N. Saad, and F. Meriaudeau, “Compressed VFH descriptor for 3D object classification,” in *3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video*, 2014, pp. 1–4.

[108] J. Liebelt, C. Schmid, and K. Schertler, “Viewpoint-independent object class detection using 3D Feature Maps,” *IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 64, no. 6, pp. 1–8, 2008.

[109] A. Ion, G. Peyré, W. G. Kropatsch, N. M. Artner, S. B. L. Mármol, and L. Cohen, “3D shape matching by geodesic eccentricity,” *2008 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work. CVPR Work.*, 2008.

[110] S. H. Kasaei, A. M. Tomé, L. S. Lopes, and M. Oliveira, “GOOD: A Global Orthographic Object Descriptor for 3D Object Recognition and Manipulation,” *Pattern Recognit. Lett.*, p. , 2016.

[111] A. Petrelli and L. Di Stefano, “On the repeatability of the local reference frame for partial shape matching,” in *2011 International Conference on Computer Vision*, 2011, pp. 2244–2251.

[112] S. Salti, F. Tombari, and L. Di Stefano, “SHOT: Unique signatures of histograms for surface and texture description,” *Comput. Vis. Image Underst.*, vol. 125, pp. 251–264, Aug. 2014.

[113] J. Yang, Z. Cao, and Q. Zhang, “A fast and robust local descriptor for 3D point cloud registration,” *Inf. Sci. (Ny).*, vol. 346–347, no. August, pp. 163–179, 2016.

[114] K. Mikolajczyk and C. Schmid, “Performance evaluation of local descriptors.,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–30, Oct. 2005.

[115] A. Aldoma, F. Tombari, L. Di Stefano, and M. Vincze, “A Global Hypothesis Verification Framework for 3D Object Recognition in Clutter,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 7, pp. 1383–1396, Jul. 2016.

[116] Y. Guo, J. Wan, M. Lu, and W. Niu, “A parts-based method for articulated target recognition in laser radar data,” *Opt. - Int. J. Light Electron Opt.*, vol. 124, no. 17, pp. 2727–2733, Sep. 2013.

[117] R. B. Rusu, N. Blodow, and M. Beetz, “Fast Point Feature Histograms

(FPFH) for 3D registration," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 3212–3217.

[118] A. Del Bimbo and P. Pala, "Content-based retrieval of 3D models," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 2, no. 1, pp. 20–43, Feb. 2006.

[119] L. J. Skelly and S. Sclaroff, "Improved feature descriptors for 3D surface matching," *SPIE Conf. 2- 3- D methods Insp. Metrol.*, vol. 6762, pp. 63–85, Sep. 2007.

[120] F. Stein, S. Member, and G. Medioni, "Structural Indexing : Efficient 3-D Object Recognition," vol. 14, no. 2, 1992.

[121] C. S. Chua and R. Jarvis, "Point Signatures: A New Representation for 3D Object Recognition," *Int. J. Comput. Vis.*, vol. 25, no. 1, pp. 63–85, 1997.

[122] Y. Sun and M. A. Abidi, "Surface matching by 3D point's fingerprint," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, 2013, vol. 2, pp. 263–269.

[123] S. Ruiz-Correa, L. G. Shapiro, and M. Melia, "A new signature-based method for efficient 3-D object recognition," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 2001, vol. 1, p. I-769-I-776.

[124] S. M. Yamany and A. A. Farag, "Surface signatures: an orientation independent free-form surface representation scheme for the purpose of objects registration and matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1105–1120, Aug. 2002.

[125] X. Li and I. Guskov, "Multi-scale Features for Approximate Alignment of Point-based Surfaces.pdf," *Sgp'05*, pp. 187–196, 2005.

[126] A. S. Mian, M. Bennamoun, and R. A. Owens, "A Novel Representation and Feature Matching Algorithm for Automatic Pairwise Registration of Range Images," *Int. J. Comput. Vis.*, vol. 66, no. 1, pp. 19–40, Jan. 2006.

[127] S. Malassiotis and M. G. Strintzis, “Snapshots: A Novel Local Surface Descriptor and Matching Algorithm for Robust 3D Surface Alignment,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 7, pp. 1285–1290, Jul. 2007.

[128] B. Taati and M. Greenspan, “Local shape descriptor selection for object recognition in range data,” *Comput. Vis. Image Underst.*, vol. 115, no. 5, pp. 681–694, May 2011.

[129] J. Novatnack and K. Nishino, “Scale-Dependent/Invariant Local 3D Shape Descriptors for Fully Automatic Registration of Multiple Sets of Range Images,” in *ECCV ’08 Proceedings of the 10th European Conference on Computer Vision: Part III*, 2008, vol. 5304, pp. 440–453.

[130] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, “Aligning point cloud views using persistent feature histograms,” *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS*, pp. 3384–3391, 2008.

[131] J. Hu and J. Hua, “Salient spectral geometric features for shape matching and retrieval,” *Vis. Comput.*, vol. 25, no. 5, pp. 667–675, 2009.

[132] J. Sun, M. Ovsjanikov, and L. Guibas, “A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion,” *Comput. Graph. Forum*, vol. 28, no. 5, pp. 1383–1392, Jul. 2009.

[133] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, “Surface feature detection and description with applications to mesh matching,” *2009 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work. CVPR Work. 2009*, pp. 373–380, 2009.

[134] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool, “Hough transform and 3D SURF for robust three dimensional classification,” *Comput. Vis. – ECCV 2010 Lect. Notes Comput. Vis.*, vol. 6316, pp. 589–602, 2010.

[135] H. Van Nguyen, F. Porikli, and M. Electric, “Concentric Ring Signature (CORS) for 3D Object Detection , Recognition , and Registration,” pp. 1–24.

[136] F. Tombari, S. Salti, and L. Di Stefano, “A combined texture-shape descriptor for enhanced 3D feature matching,” in *2011 18th IEEE International Conference on Image Processing*, 2011, pp. 809–812.

[137] Z. Marton, D. Pangercic, N. Blodow, J. Kleinehellefort, and M. Beetz, “General 3D modelling of novel objects from a single view,” in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 3700–3705.

[138] I. Kokkinos, M. M. Bronstein, R. Litman, and a. M. Bronstein, “Intrinsic shape context descriptors for deformable shapes,” *2012 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 159–166, Jun. 2012.

[139] T. Fiolka, J. Stückler, D. Klein, D. Schulz, and S. Behnke, “Sure: surface entropy for distinctive 3d features,” *Proc. Spat. Cogn. 2012*, no. September, 2012.

[140] F. Sukno, J. Waddington, and P. F. Whelan, “Rotationally Invariant 3D Shape Contexts using Asymmetry Patterns,” in *Proceedings of the International Conference on Computer Graphics Theory and Applications and International Conference on Information Visualization Theory and Applications*, 2013, pp. 7–17.

[141] S. A. A. Shah, M. Bennamoun, F. Boussaid, and A. a. El-Sallam, “A Novel Local Surface Description for Automatic 3D Object Recognition in Low Resolution Cluttered Scenes,” in *2013 IEEE International Conference on Computer Vision Workshops*, 2013, pp. 638–643.

[142] H. Zeng, R. Zhang, and M. Huang, “Improved 3D Local Feature Descriptor Based on Rotational Projection Statistics and Depth Information,” in *Computer Vision: CCF Chinese Conference, CCCV 2015, Xi’an, China, September 18-20, 2015, Proceedings, Part II*, H. Zha, X. Chen, L. Wang, and Q. Miao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 139–147.

[143] S. M. Prakhya, B. Liu, and W. Lin, “B-SHOT: A binary feature descriptor for

fast and efficient keypoint matching on 3D point clouds," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 1929–1934.

[144] M. Zhen, W. Wang, and R. Wang, "Signature of unique angles Histograms for 3D data description," in *Multimedia Expo Workshops (ICMEW), 2015 IEEE International Conference on*, 2015, pp. 1–6.

[145] B. Lin, F. Zhao, T. Tamaki, F. Wang, and L. Xiao, "SIPF: Scale invariant point feature for 3D point clouds," in *Image Processing (ICIP), 2015 IEEE International Conference on*, 2015, pp. 2611–2615.

[146] T.-W. R. Lo and J. P. Siebert, "Local feature extraction and matching on range images: 2.5D SIFT," *Comput. Vis. Image Underst.*, vol. 113, no. 12, pp. 1235–1250, Dec. 2009.

[147] E. R. Do Nascimento, G. L. Oliveira, A. W. Vieira, and M. F. M. Campos, "On the development of a robust, fast and lightweight keypoint descriptor," *Neurocomputing*, vol. 120, pp. 141–155, 2013.

[148] J. Krizaj, V. Struc, and F. Mihelic, "A feasibility study on the use of binary keypoint descriptors for 3D face recognition," in *In Mexican Conference on Pattern Recognition*, 2014, vol. 8495 LNCS, pp. 142–151.

[149] J. J. Koenderink, *Solid Shape*. Cambridge, MA: MIT Press, 1990.

[150] Xiaoguang Lu, A. Jain, and D. Colbry, "Matching 2.5D face scans to 3D models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 31–43, Jan. 2006.

[151] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *2003 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, 2003. Proceedings.*, vol. 2, p. II-257-II-263.

[152] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 105–119, 2010.

[153] E. Mair, G. D. Hager, D. Burschka, M. Suppa, and G. Hirzinger, “Adaptive and Generic Corner Detection Based on the Accelerated Segment Test,” in *Computer Vision -- ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part II*, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 183–196.

[154] X. Yu and T. Huang, “A SIFT-based Image fingerprinting approach robust to geometric transformations,” in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium*, 2009, pp. 1665–1668.

[155] V. Seib, M. Kusenbach, S. Thierfelder, and D. Paulus, “Object recognition using Hough transform clustering of SURF features,” in *Scientific Cooperations International Workshops on Electrical and Computer Engineering Subfields*, 2014.

[156] M. a Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[157] S. Katz, A. Tal, and R. Basri, “Direct visibility of point sets,” *ACM Trans. Graph.*, vol. 26, no. 3, p. 24, Jul. 2007.

[158] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, “The Princeton shape benchmark,” in *International Conference on Shape Modeling and Applications*, 2004.

[159] “Grabcad.” [Online]. Available: https://grabcad.com/. [Accessed: 19-Feb-2015].

[160] D. V. Rabinkin, E. Rutledge, and P. Monticciolo, “Missile signal processing common computer architecture for rapid technology upgrade,” in *Proc. SPIE 5559, Advanced Signal Processing Algorithms, Architectures, and Implementations XIV*, 2004, pp. 131–145.

[161] T. Andrews, “Computation Time Comparison Between Matlab and C++ Using Launch Windows,” *Aerosp. Eng.*, pp. 1–6, 2012.

[162] S. B. Aruoba and J. Fernández-Villaverde, “A Comparison of Programming Languages in Economics,” Cambridge, MA, Jun. 2014.

[163] “primatelabs.” [Online]. Available: http://browser.primatelabs.com/. [Accessed: 21-Feb-2016].

[164] M. Corsini, P. Cignoni, and R. Scopigno, “Efficient and flexible sampling with blue noise properties of triangular meshes.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 6, pp. 914–24, Jun. 2012.

[165] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and J. Zhang, “Performance Evaluation of 3D Local Feature Descriptors,” in *Computer Vision -- ACCV 2014*, 2015, pp. 178–194.

[166] J. Bauer, N. Sunderhauf, and P. Peter, “Comparing several implementations of two recently published feature detectors,” in *6th IFAC Symposium on Intelligent Autonomous Vehicles*, 2007, vol. 154, no. 3, pp. 143–148.

[167] A. Zaharescu, E. Boyer, and R. Horaud, “Keypoints and local descriptors of scalar functions on 2D manifolds,” *Int. J. Comput. Vis.*, vol. 100, no. 1, pp. 78–98, 2012.

[168] A. Kim, O. Gwun, and J. Song, “Extraction of 3D Feature Descriptor Using the Distribution of Normal Vectors,” in *International Conference on Flexible Query Answering Systems*, 2009, pp. 191–200.

[169] L. A. Alexandre, “3D Descriptors for Object and Category Recognition : a Comparative Evaluation,” *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 34, no. 8, pp. 1–6, Aug. 2012.

[170] J. Maye and S. Dahinden, “Feature-based Extrinsic Calibration of Camera and 3D Laser Range Finder,” Swiss Federal Institute of Technology Zurich, 2010.

[171] T.-H. Yu, O. J. Woodford, and R. Cipolla, “A Performance evaluation of volumetric 3D interest point detectors,” *Int. J. Comput. Vis.*, vol. 102, no. 1–

3, pp. 180–197, Sep. 2012.

[172] K. Mikolajczyk and C. Schmid, “Scale & Affine Invariant Interest Point Detectors,” *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63–86, 2004.

[173] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-Up Robust Features (SURF),” no. September, 2008.

[174] D. Bekele, M. Teutsch, and T. Schuchert, “Evaluation of binary keypoint descriptors,” in *2013 IEEE International Conference on Image Processing*, 2013, pp. 3652–3656.

[175] A. Aldoma *et al.*, “Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation,” *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 80–91, Sep. 2012.

[176] M. Ebrahimi and W. W. Mayol-Cuevas, “SUSurE: Speeded Up Surround Extrema feature detector and descriptor for realtime applications,” *2009 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, pp. 9–14, Jun. 2009.

[177] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, “KAZE features,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012, vol. 7577 LNCS, no. PART 6, pp. 214–227.

[178] B. F. Fraundorfer and D. Scaramuzza, “Visual Odometry,” no. June, 2012.

[179] J. Heinly, E. Dunn, and J. Frahm, “Comparative Evaluation of Binary Features.”

[180] E. Hsiao, A. Collet, and M. Hebert, “Making specific features less discriminative to improve point-based 3D object recognition,” *2010 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 2653–2660, Jun. 2010.

[181] M. H. Lee and I. K. Park, “Robust feature description and matching using local graph,” *2013 Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit*

Conf., pp. 1–4, Oct. 2013.

[182] P. Moreels and P. Perona, “Evaluation of Features Detectors and Descriptors based on 3D objects 1 Introduction.”

[183] J.-M. Morel and G. Yu, “ASIFT: A New Framework for Fully Affine Invariant Image Comparison,” *SIAM J. Imaging Sci.*, vol. 2, no. 2, pp. 438–469, Jan. 2009.

[184] X. Qu, F. Zhao, M. Zhou, and H. Huo, “A Novel Fast and Robust Binary Affine Invariant Descriptor for Image Matching,” *Math. Probl. Eng.*, vol. 2014, pp. 1–7, 2014.

[185] A. Redondi, L. Baroffio, J. Ascenso, M. Cesana, and M. Tagliasacchi, “RATE-ACCURACY OPTIMIZATION OF BINARY DESCRIPTORS Dipartimento di Elettronica e Informazione , Politecnico di Milano ˜ es , Lisbon Instituto Superior de Engenharia de Lisboa - Instituto de Telecomunicac ¸ o,” no. 296676, pp. 2910–2914, 2013.

[186] G. Clark, “Probability Density and CFAR Threshold Estimation for Hyperspectral Imaging,” California, 2004.

[187] J. Daugman, “Probing the uniqueness and randomness of iriscodes: Results from 200 billion iris pair comparisons,” in *Proceedings of the IEEE*, 2006, vol. 94, no. 11, pp. 1927–1934.

[188] W. Wohlkinger, A. Aldoma, R. B. Rusu, and M. Vincze, “3DNet: Large-scale object class recognition from CAD models,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 5384–5391, 2012.

[189] A. Aldoma, “DAGM dataset,” 2012. [Online]. Available: http://users.acin.tuwien.ac.at/aaldoma/datasets/DAGM.zip.

[190] K. Lai, L. Bo, X. Ren, and D. Fox, “RGB-D Object Recognition: Features, Algorithms, and a Large Scale Benchmark,” in *Consumer Depth Cameras for Computer Vision*, London: Springer London, 2013, pp. 167–192.

[191] R. B. Rusu, A. Holzbach, M. Beetz, and G. Bradski, “Detecting and

segmenting objects for mobile manipulation," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference*, 2009, pp. 47–54.

[192] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Proceedings Shape Modeling Applications, 2004.*, pp. 167–388.

[193] D. Palossi, F. Tombari, S. Salti, M. Ruggiero, L. Di Stefano, and L. Benini, "GPU-SHOT: Parallel Optimization for Real-Time 3D Local Description," in *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 584–591.

[194] Y. Lei, M. Bennamoun, and A. A. El-Sallam, "An efficient 3D face recognition approach based on the fusion of novel local low-level features," *Pattern Recognit.*, vol. 46, no. 1, pp. 24–37, Jan. 2013.

[195] S. Filipe and L. a. Alexandre, "A Comparative Evaluation of 3D Keypoint Detectors in a RGB-D Object Dataset," in *Proceedings of the 9th International Conference on Computer Vision Theory and Applications*, 2014, pp. 476–483.

[196] A.-E. Ichim, "PFHRGB." [Online]. Available: http://www.pointclouds.org/blog/gsoc/aichim/index.php. [Accessed: 20-Aug-2016].

[197] A. S. Mian, "Calculating Spin Images." [Online]. Available: http://staffhome.ecm.uwa.edu.au/~00053650/code.html. [Accessed: 18-Aug-2016].

[198] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96*, 1996, pp. 303–312.

[199] F. Tombari, "Keypoints and Features," *CGLibs presentation*, 2013. [Online]. Available:

http://www.pointclouds.org/assets/uploads/cglibs13_features.pdf.

[200] R. B. Rusu and S. Cousins, “3D is here: Point Cloud Library (PCL),” in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1–4.

[201] MATLAB, “MATLAB Central FileExchange.” [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/. [Accessed: 20-Apr-2014].

[202] F. Pomerleau, F. Colas, and R. Siegwart, “A Review of Point Cloud Registration Algorithms for Mobile Robotics,” *Found. Trends Robot.*, vol. 4, no. 1, pp. 1–104, 2015.

[203] L. Gagnon and R. Klepko, “Hierarchical Classifier Design for Airborne SAR Images of Ships,” in *Aerospace/Defense Sensing and Controls*, International Society for Optics and Photonics, 1998, pp. 38–49.

[204] E. Blasch, “Fusion of HRR and SAR information for Automatic Target Recognition and Classification,” in *International Conference on Informantion Fusion*, 1999, pp. 1221–1227.

[205] H. Zhang, N. M. Nasrabadi, Y. Zhang, and T. S. Huang, “Multi-view automatic target recognition using joint sparse representation,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2481–2497, 2012.

[206] L. Alexandre, “3D descriptors for object and category recognition: a comparative evaluation,” *Work. Color. Camera Fusion Robot. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 1, no. 3, 2012.

[207] G. Berk and L. Akarun, “Comparative Analysis of Decision-level Fusion Algorithms for 3D Face Recognition,” pp. 104–107, 2006.

[208] P. J. Besl and N. D. McKay, “A Method for Registration of 3D Shapes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 586–606, 1992.

[209] Presagis, “OpenFlight visual simulation.” [Online]. Available: http://www.presagis.com/products_services/standards/openflight/.

[Accessed: 01-Aug-2016].

[210] R. J. Campbell and P. J. Flynn, “A Survey Of Free-Form Object Representation and Recognition Techniques,” *Comput. Vis. Image Underst.*, vol. 81, no. 2, pp. 166–210, Feb. 2001.

[211] F. Dios, A. Rodrigues, and A. Comeron, “Scintiallation and beam-wander analysis in an optical ground station-satellite uplink,” *Appl. Opt.*, vol. 43, no. 19, pp. 3866–3873, 2004.

[212] D. C. Soreide, R. K. Bogue, L. J. Ehernberger, S. M. Hannon, and D. A. Bowdle, “Airborne coherent LIDAR for advanced in-flight measurements,” *Proc. Conf. Aviat. Range, Aerosp. Meteorol.*, pp. 1–8, 2000.

[213] S. Gokturk, H. Yalcin, and C. Bamji, “A time-of-flight depth sensor-system description, issues and solutions,” *Comput. Vis. Pattern …*, vol. 0, no. C, pp. 35–35, 2004.

[214] C. Liu, R. Szeliski, S. B. Kang, L. Zitnick, and W. Freeman, “Automatic Estimation and Removal of Noise from a Single Image,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 299–314, 2008.

[215] A. Foi, M. Trimeche, V. Katkovnik, and K. Egazarian, “Practical Poissonian-Gaussian Noise Modeling and Fitting for Single-Image Raw-Data,” *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1737–1754, 2008.

[216] J. Rice, *Mathematical statistics and data analysis*, 2nd ed. Belmond, California: Duxbury Press, 1995.

[217] S. Hasinoff, “Photon, Poisson Noise,” in *Computer Vision, A reference Guide*, 2014, pp. 608–610.

[218] A. P. French and E. F. Taylor, *An introduction to Quantum Physics*. London: Van Nostrand Reinhold, 1978.

APPENDIX A – Processing Time

One of the main parameters of an algorithm is its computational efficiency. In this appendix, the processing threshold limit will be determined under the scope of 3D ATR for missile applications. For the purpose of this research, the following assumptions are be made:

a. This research is a feasibility study on 3D ATR for missile platforms rather than a complete ready-to-use missile 3D ATR solution. Hence, an approximate estimation of the real-time processing requirement is sufficient.
b. This research focuses purely on target recognition and therefore during the calculation of the afforded 3D ATR processing latency, the missile's speed will be simulated in a simplistic manner. This is important because otherwise, this thesis will deviate from an ATR into an aerodynamics and missile propulsion problem. The latter holds true because the missile's speed derivation per flight phase i.e. boost, sustain and terminal guidance is influenced by quite a few parameters such as aerodynamic drag depending on the missile's speed (subsonic, supersonic), flight level, environmental conditions, specific impulse of the propellant. Taking into account all these information will deviate this thesis from its score, which is 3D ATR from missile platforms.
c. Algorithms that are applied on missile platforms are developed in C/C++, while this study uses MATLAB tools. Therefore, the processing setback of MATLAB will be compensated with a speedup factor.

For the upcoming calculations, four popular anti-tank missiles (Javelin, Kornet, Brimstone and Spike-ER) and MBTs (T72, T90, M1A1 Abrams, Leopard 2A6) are considered. Since this is military equipment, the exact specifications are not available and thus open source data from the internet are used as presented in Table A-1. As already stated, these approximate values are sufficient for the purpose of this thesis.

Table A- 1 Missile and MBT target velocity

Platform type	Platform name	Speed (m/s)	Diameter/ width (m)	Length (m)
Anti-tank missile	Javelin	578	0.127	1.1
	kornet	300	0.152	1.2
	Brimstone	450	0.178	1.8
	Spike-ER	200	0.110	1.5
MBT	T72	17	3.59	9.53
	M1A1 Abrams	13	3.66	9.77
	T90	17	3.78	9.63
	Leopard 2A6	20	3.75	9.97

Since this research focuses on the ATR concepts rather than the exact flight profile of the missile, a simplified motion is considered. In specific, for a timeframe Δt the missile's trajectory and the MBT's course are simplified having a linear route at constant velocity and at right angles. It is assumed that at time t_0 the MBT is at the centreline axis of the missile. A graphical representation is shown in Figure A-1.

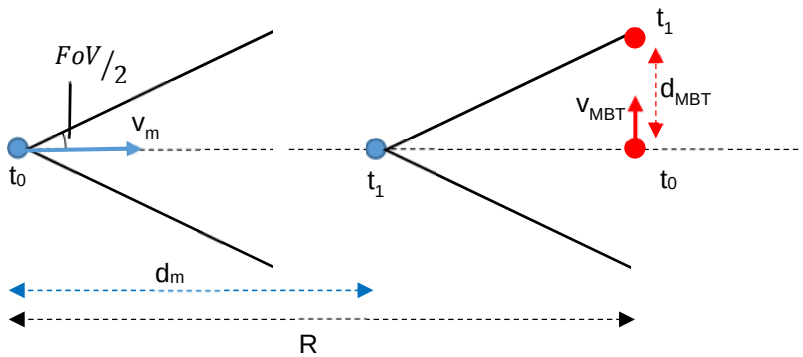


Figure A- 1 Simple missile (blue dot) vs. MBT (red dot) engagement scenario for the timeframe $\Delta t=t_0-t_1$

where the blue dot is the missile, the red dot is the MBT, d_m and d_{MBT} are the distances travelled during the $\Delta t = t_0 - t_1$ timeframe, V_m and V_{MBT} the missile's and MBT's vectorial velocities, FoV is the LIDAR optics Field of View and *R* the missile – MBT range at time t_0.

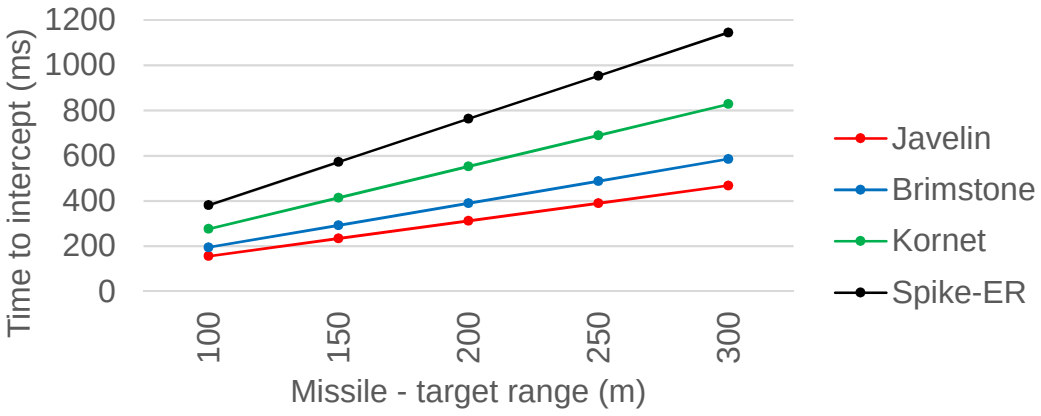
Given that the target is at the LIDAR's boresight, aim of this scenario is to determine the time the missile can afford until the MBT exits the seeker's half-FoV. For the scenario of Figure A-1:

$$\tan(FoV/2) = \frac{d_{MBT}}{R - d_m} = \frac{V_{MBT}\Delta t}{R - V_m \Delta t} \quad \textbf{(A- 1)}$$

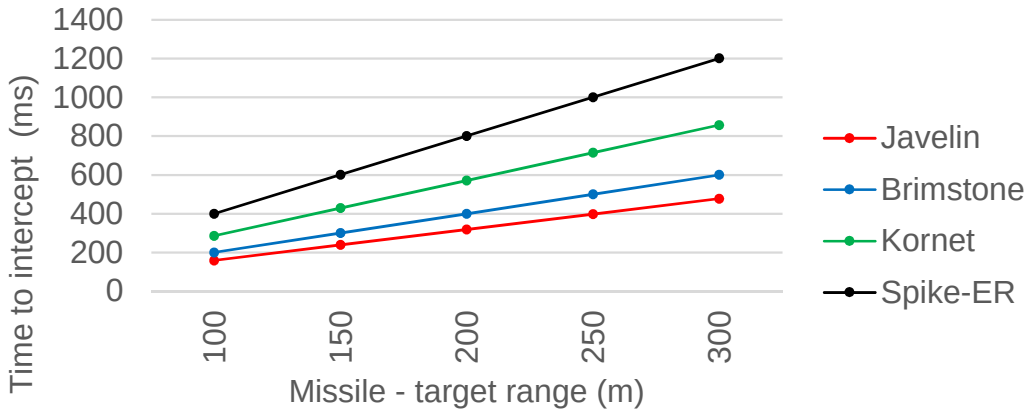
$$\Delta t = \frac{R \cdot \tan(FoV/2)}{V_{MBT} + V_m \cdot \tan(FoV/2)} \quad \textbf{(A- 2)}$$

During these experiments, a FoV of 30° is considered. This is because, current commercial LIDAR devices usually have a 360° Horizontal FoV and 30° vertical FoV. For missile applications though, the Horizontal FoV has to be constrained within a few degrees, as the missile body frame prohibits a full 360° coverage. Thus, it is reasonable to assume that a 30° horizontal FoV is a good compromise as it is currently used as a Vertical FoV and affords a good target detection/ recognition probability. The latter can be explained as, the larger the FoV the higher the probability for the MBT to be within the FoV and thus being detected and ultimately recognised.

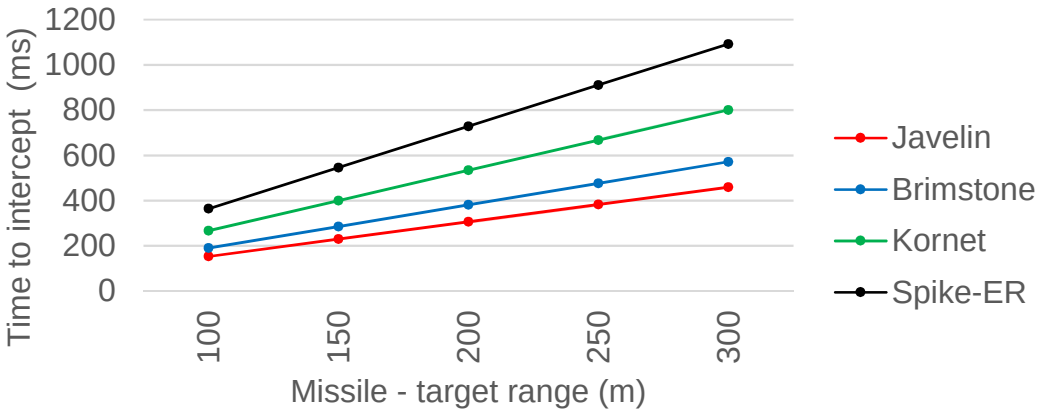
Based on the aforementioned assumptions, Figure A-2 shows the time to intercept i.e. time until the missile reaches the MBT target, vs. the missile – target range. As expected the closer the missile to the target is, the smaller the time to intercept and thus the faster the 3D ATR has to be. It should be noted that the timings presented consider a single processing loop of the algorithm, while for robustness and in order to increase the recognition rate, at least two sequential positive target recognition should take place. This secures with a high probability that the target within the scene is indeed the one classified by the ATR algorithm with a high probability. Therefore, the timings presented in Figure A-2 have to be divided by a factor of two.



(a)



(b)



(c)

Figure A- 2 Time to intercept vs. missile – target range per common anti0tank missile and MBT combination (a) T72/ T90 (b) M1A1 Abrams (c) Leopard 2A6

The Leopard 2A6 MBT has the highest speed of 17m/s, while the javelin missile has the highest velocity among the anti-tank missiles examined, providing a time to intercept of only 153/2=76.5ms at 100-meters missile-target range.

It is worth noting that missile oriented algorithms are implemented in C/C++ to gain computational efficiency and real-time performance. As stated in Section 1.6, this research relies on MATLAB coding. Although MATLAB is not as processing efficient as C/C++ is, it is a widely used algorithm prototyping platform that fulfils the 3D ATR performance recognition rate requirements. Since this research does not aim at providing a readily available solution, but is rather a feasibility study of innovative concepts, the processing setback of MATLAB will be compensated with a speedup factor. The latter is selected such as on the PC's used and in a MATLAB environment, the processing time threshold of the 3D ATR algorithms is 500ms.

APPENDIX B – Stereo Vision

From a set of images obtained from a stereo camera it is possible to create 3D data. In that case the depth Z is given by:

$$Z = \frac{f \cdot B}{d} \qquad \textbf{(B- 1)}$$

where f is the focal length, B the baseline distance i.e. the distance between the two cameras and d the disparity. The latter is the distance between two corresponding pixels between the left and the right camera image of the stereo camera configuration, measured in pixels. The further away an object from the stereo camera is, the smaller the disparity. Hence, the maximum depth that can be estimated is for disparity equal to one pixel shift.

Based on Equation B-1, various commonly used focal length values (3mm – 15mm), 80% of the diameter of the popular anti-tank missiles as presented in Table A-1 and a disparity of one pixel, Figure B-1 presents the maximum that can be estimated. Objects that are further away are considered as background and their depth cannot be estimated. For completeness, it is worth mentioning that a missile's diameter is measured from the outer casing and thus on average only an 80% of that diameter is available to host the missile's hardware.

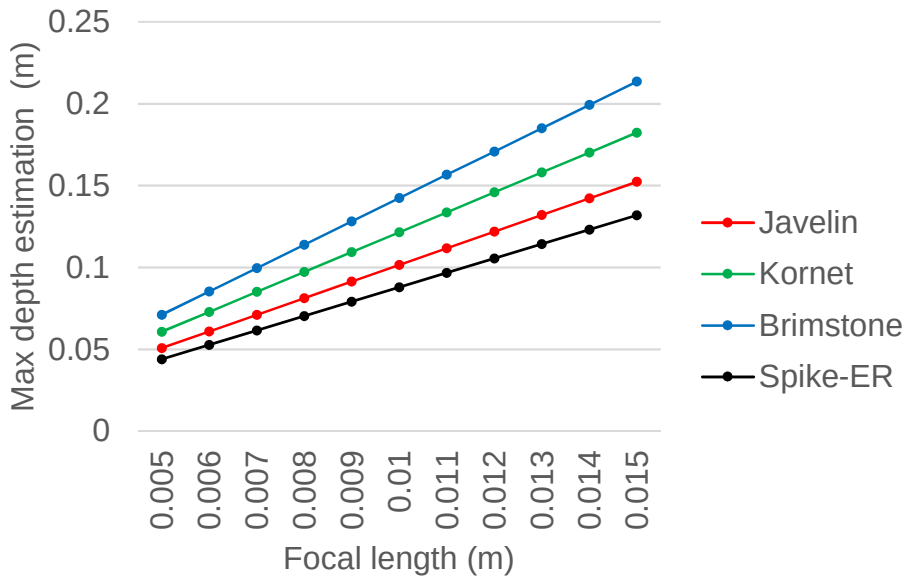


Figure B- 1 Maximum depth estimation for current popular anti-tank missiles

From Figure B-1 it is obvious that for the Brimstone, which is has the largest diameter of the missiles evaluated, it can afford a depth estimation of only 0.21 meters. That depth is not acceptable as too low and thus the stereoscopic 3D data construction is not applicable for missile type applications.

APPENDIX C – Atmospheric Noise Simulation

This research considers the case of a Time-of-Flight LIDAR type that uses a typical military laser operating at 1000nm. The LIDAR sensor suffers from various noise sources such as thermal noise and background noise with the latter being of random shot type i.e. follows a random Poisson distribution [78], [212], [213]. However, if the number of photons enclosed in a laser burst is excessive, then LIDAR's shot noise can be simulated via a Gaussian distribution with variance that depends on the photon count [214], [215]. This is because for a small amount of photons within each laser burst, shot noise is accurately modelled by a Poisson distribution, while for large number of photons, the central limit theorem [216] ensures that the Poisson distribution approaches a Gaussian [217]. Aim of this Appendix is to demonstrate that typical ToF LIDARs have an excessive number of photons per laser burst.

Given that the operating wavelength of a typical military laser is 1064mn the corresponding frequency is:

$$v = \frac{c}{\lambda} = \frac{3 \cdot 10^{8}\ \text{m/s}}{1000 \cdot 10^{-9}\ \text{m}} = 3 \cdot 10^{14}\ \text{Hz} \qquad \textbf{(C- 1)}$$

It should be noted that the true speed of light depends on several parameters related to the medium it propagates through. Despite that, for simplicity, the speed of light at vacuum is used throughout calculations.

The Planck – Einstein relation [218] states that:

$$E = h \cdot v \qquad \textbf{(C- 2)}$$

where E is the energy of a single photon, h Planck's constant and v the frequency of the photon. Combining Equations C-1 and C-2, the energy per photon is calculated:

$$E_{photon} = 6.626 \cdot 10^{-34} Js \cdot 3 \cdot 10^{14} s^{-1} = 19.878 \cdot 10^{-20} J \qquad \textbf{(C- 3)}$$

A current commercial type small sized LIDAR [82] consumes 8W per pulse thus:

$$E_{pulse} = \frac{power}{PRF} = \frac{8W}{300KHz} = 26\mu J \qquad \textbf{(C- 4)}$$

with each pulse containing:

$$26 \cdot 10^{-6} J \cdot \frac{1\,photon}{19.878 \cdot 10^{-20} J} = 1.3079 \cdot 10^{14}\ photons \qquad \textbf{(C- 5)}$$

Even in the extreme case of 1J laser energy per pulse the number of photons becomes:

$$1J \cdot \frac{1\,photon}{19.878 \cdot 10^{-20} J} = 5.0307 \cdot 10^{18}\ photons \qquad \textbf{(C- 6)}$$

In any case, the number of photons per pulse is extremely large and thus the central limit theorem is applicable ensuring that the Poisson distribution approaches a Gaussian. Therefore, in all noise trials of this thesis, atmospheric noise is simulated by Gaussian noise. An advantage of adding Gaussian noise is allowing a direct comparison with current 3D object recognition approaches that explicitly use Gaussian noise. For the sake of comparison reasons with current literature, the variance/ standard deviation of the Gaussian distribution is disconnected from the photon count, and depends on the average point cloud resolution which is the norm for the computer vision literature.